Characterization of flat outputs for the diagnostic of integer or non-integer systems : application for the diagnostic of a hydraulic system and a thermal system

Rim Rammal

THÈSE PRÉSENTÉE
POUR OBTENIR LE GRADE DE

DOCTEUR DE
L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE n°209 : Sciences Physiques et de l'Ingénieur

SPÉCIALITÉ : Automatique, Productique, Signal et Image, Ingénierie Cognitique

Par **Rim RAMMAL**

Caractérisation des sorties plates pour le diagnostic de systèmes entiers ou non entiers : application pour le diagnostic d'un système hydraulique et d'un système thermique

Soutenue le 15 janvier 2021

Membres du jury :

M. PONSART, Jean-Christophe	Professeur	CRAN, Université de Lorraine	Rapporteur
Mme MAAMRI-TRIGEASSOU, Nezha	MCF-HDR	LIAS, ENSIP, Université de Poitiers	Rapporteure
M. LÉVINE, Jean	DR Emérite	CAS, Ecole des Mines de Paris	Président
M. OLLIVIER, François	CR CNRS	LIX, Ecole polytechnique	Examinateur
M. MELCHIOR, Pierre	Professeur	IMS, Bordeaux INP	Co-Directeur
M. CAZAURANG, Franck	Professeur	IMS, Université de Bordeaux	Co-Directeur
M. AIRIMITOAIE, Tudor-Bogdan	MCF	IMS, Université de Bordeaux	Co-Encadrant

IMS - UMR 5218 CNRS, Université de Bordeaux/Bordeaux INP

351 Cours de la Libération, 33405 Talence CEDEX, France

Caractérisation des sorties plates pour le diagnostic de systèmes entiers ou non entiers : application pour le diagnostic d'un système hydraulique et d'un système thermique

Résumé

La platitude différentielle est une propriété des systèmes dynamiques qui permet la transformation d'un système très complexe en un système plus simple appelé système plat. On dit qu'un système dynamique est plat si, et seulement si, il existe un vecteur, appelé vecteur de sortie plate, formé par les variables d'état et d'entrée du système, tel que tous les états, entrées et sorties du système peuvent être exprimés en fonction de ce nouveau vecteur et de ses dérivées temporelles successives. La platitude différentielle a de nombreuses applications dans la théorie du contrôle automatique, telles que la planification des trajectoires, le suivi des trajectoires et la conception de contrôleurs robustes. De plus, la propriété de platitude est récemment entrée dans le domaine de la détection et de l'isolation des défauts. La détection et l'isolation des défauts sont un sous-domaine de l'ingénierie de contrôle automatique qui traite de la surveillance d'un système, de l'identification du moment où un défaut s'est produit, et de la détermination du type de défaut et de sa localisation. La détection des défauts est effectuée en analysant la différence entre les mesures des capteurs et des actionneurs et les valeurs attendues, dérivées de n'importe quel modèle et appelées valeurs redondantes. Il est courant de dire qu'une erreur est détectée si l'écart ou le résidu dépasse un certain seuil prédéfini. L'isolation des défauts, à son tour, doit permettre de localiser le défaut dans la machine. La méthode la plus récente de détection et d'isolation des défauts, basée sur la propriété de la platitude, calcule des variables redondantes à partir de la mesure de la sortie plate du système et de ses dérivées temporelles successives. Ensuite, des résidus sont déduits de la différence entre les variables mesurées et les variables redondantes. La détection des défauts par cette méthode est garantie. Cependant,

l'utilisation d'une seule sortie plate ne permet pas, dans certains cas, d'isoler certains défauts. L'idée proposée par les développeurs de la méthode était d'utiliser plusieurs sorties plates pour augmenter le nombre de résidus, ce qui augmenterait les chances d'isoler davantage de défauts. Cependant, il a également été remarqué que le choix de ces sorties plates n'est pas arbitraire. En d'autres termes, il existe des sorties plates qui, lorsqu'elles sont utilisées ensemble, augmentent l'isolabilité des défauts et d'autres qui ne le font pas. Un des objectifs de ce manuscrit est de caractériser les sorties plates afin d'obtenir une meilleure isolabilité des défauts. Cette caractérisation est ensuite vérifiée par des simulations et des expériences sur un système hydraulique, le système des trois cuves.

Au cours de la dernière décennie, de nombreuses études ont montré qu'il existe des systèmes tels que les systèmes thermiques, les systèmes viscoélastiques et les systèmes chimiques qui peuvent être modélisés par des équations différentielles fractionnaires. Par conséquent, les méthodes classiques de détection et d'isolation des défauts, développées à l'origine pour traiter les systèmes d'ordre entier, ne convenaient pas aux systèmes d'ordre fractionnaire, et des méthodes de détection et d'isolation des défauts spécifiques aux systèmes d'ordre fractionnaire ont dû être développées. Un deuxième objectif de ce manuscrit est d'étendre la caractérisation des sorties plates, proposée pour la classe des systèmes plats d'ordre entier à la classe des systèmes plats linéaires d'ordre fractionnaire, puis d'appliquer cette caractérisation à la détection et à l'isolation des défauts qui peuvent apparaitre sur les capteurs et les actionneurs de ces systèmes. L'efficacité de cette caractérisation est également vérifiée par des simulations sur un système thermique bi-dimensionnel.

Mots clés : Platitude différentielle, sortie plate, système non linéaire, système fractionnaire, détection et isolation de défaut.

Characterization of flat outputs for the diagnostic of integer or non-integer systems: application for the diagnostic of a hydraulic system and a thermal system

Abstract

The differential flatness is a property of dynamic systems that allows the transformation of a very complex system into a simpler one called flat system. Roughly speaking, a dynamic system is said to be flat if, and only if, there exists a vector, called flat output vector and formed by the state and input variables, such that all the system states, inputs and outputs can be expressed in function of this new vector and its successive time derivatives. The differential flatness property has many applications in automatic control theory, such as trajectory planning, trajectory tracking and the designing of robust controllers. Moreover, the flatness property has recently entered the field of fault detection and isolation. In short, fault detection and isolation is a sub-domain of automatic control engineering that deals with monitoring a system, identifying when a fault has occurred, and determining the type of fault and its location. Fault detection is performed by analyzing the difference between sensor and actuator measurements and their expected values, derived from any model and called redundant values. It is common to say that an error is detected if the deviation or residue exceeds a certain predefined threshold. Fault isolation, in turn, must make it possible to locate the fault in the machine. The most recent method of fault detection and isolation, based on the flatness property, calculates redundant variables from the measurement of the flat output of the system and its successive time derivatives. Then, the residues are deduced from the difference between the measured variables and the redundant variables. Fault detection by this method is guaranteed. However, the use of a single flat output does not allow, in some cases, to isolate some faults. The idea proposed by the developers of the method was to use several flat outputs to increase the number of the

residual signals, which would increase the chances of isolating more faults. However, it was also noticed that the choice of these flat outputs is not arbitrary. That is, there are flat outputs that, when used together, increase the isolability of faults and others that do not. One of the objectives of this manuscript is to characterize the flat outputs in order to obtain a better fault isolability. This characterization is then verified by simulations and experiments on a hydraulic system, the three-tank system.

Over the last decade, numerous studies have shown that there are systems such as thermal systems, viscoelastic systems and chemical systems that can be modeled by fractional differential equations. Therefore, classical methods of fault detection and isolation, originally developed to deal with integer order systems, were not suitable for fractional order systems, and fault detection and isolation methods specific to fractional order systems had to be developed. A second objective of this manuscript is to extend the characterization of flat outputs, proposed for the class of integer order flat systems to the class of fractional order linear flat systems, and then to apply this characterization to the detection and isolation of faults that may appear on the sensors and actuators of these systems. The effectiveness of this characterization is also verified by simulations on a bi-dimensional thermal system.

Keywords: Differential flatness, flat output, nonlinear system, fractional system, fault detection and isolation.

Unité de recherche

Laboratoire de l'Intégration du Matériau au Système

UMR 5218, CNRS

Université de Bordeaux

Bordeaux INP

Campus Talence, Bât A31

351 Cours de la Libération, 33405 Talence CEDEX, France

"Don't judge each day by the harvest you reap but by the seeds you plant"

Robert Louis Stevenson

"An approximate answer to the right question is worth a great deal more than a precise answer to the wrong question"

John Tukey

To my parents and my grandmother..

Acknowledgements

This thesis manuscript is the result of three years spent at the IMS Laboratory, within the CRONE and FFTG teams. This work allowed me to get involved in the domain of automatic control.

First of all, I would like to thank my supervisors Mr.Pierre Melchior, Mr.Franck Cazaurang and Mr.Tudor-Bogdan Airimitoaie who have supervised me throughout this thesis and who have shared their brilliant intuitions and ideas with me. I would also like to thank them for the trust and patience they have shown me as well as their kindness, their permanent availability and for the valuable encouragement they gave me.

I would also like to thank all the members of the CRONE and FFTG teams. The dynamics of these two teams and the good ambiance that reigns there are encouraging. Especially, I would like to thank Mr. Stéphane Victor for his kindness and the time he gave me to discuss several topics and to give me answers to several questions, and Mr. Ghazi Bel Haj Frej for his support and help during these three years and for the hours of discussion over coffee and for the best memories.

I would also like to thank the jury members, starting with Ms. Nezha Maamri-Trigeassou and Mr. Jean-Christophe Ponsart who were in charge of reporting my work. Mr. Jean Lévine who honored me by being the president of the jury, and for whom a part of this thesis is the result of a collaboration of more than two years with him. It is at his side that I understood what rigor and precision meant. Mr. François Ollivier who kindly accepted to be examiner and for the rich discussions that led to better results.

Finally, I would like to thank my family for their constant support and encouragement, and Majd my life partner for having supported me during these long years of work, for having accompanied me in all the moments of sorrow and joy and I hope to be able to do the same at his side for the years to come.

Contents

List of Figures

List of Tables

List of Abbreviations

FDI	Fault Detection and Isolation
FTC	Fault Tolerant Control
FTCS	Fault Tolerant Control System
MIMO	Multi Input Multi Output
PI	Proportional Integral
SLF	Static Linear Feedback

List of Symbols

$\mathbb{N}$	set of natural number
$\mathbb{N}^*$	set of natural number with 0 excluded: $\mathbb{N} \setminus \{0\}$
$\mathbb{R}$	set of real number
$\mathbb{C}$	set of complex number
$\mathcal{C}^\infty$	set of infinitely differentiable functions
$\mathfrak{X}$	manifold of jets of infinite order
$\mathbb{R}^m_\infty$	infinite product of $\mathbb{R}^m$: $\mathbb{R}^m \times \mathbb{R}^m \times \ldots$
$T_x X$	tangent space of manifold X at point $x \in X$
$\mathfrak{K}$	ring equipped with two binary operations $+$ and $\times$
$\mathfrak{K}[X^\gamma]$	set of polynomials in X^γ with coefficients in $\mathfrak{K}$
$\mathcal{R}$	ring of meromorphic functions
$A^{p\times q}$	set of matrices of size $p \times q$
$diag\ \{A\}$	diagonal matrix with diagonal term the matrix A
A^{-1}	inverse of the matrix A
$A^\dagger$	pseudo-inverse of the matrix A
$A^\perp$	orthonormal of the matrix A
A^T	transpose of the matrix A

$\ker(A)$	kernel of the matrix A
$rank\ (A)$	rank of the matrix A
Π	permutation matrix
I_q	identity matrix of size $q \times q$
$0_{p\times q}$	zero matrix of size $p \times q$
$\Gamma(x)$	Euler's function
sgn	sign function
$\mathrm{pr}_X(Y)$	canonical projection from Y to X
$\frac{d}{dt}$	differentiation operator
$\mathscr{L}$	Laplace transform operator
s	Laplace variable
s^γ	derivative of order γ in the Laplace domain
$\mathbf{I}^\gamma$	integral of any order γ
$\mathbf{D}^\gamma$	derivative of any order γ
$\frac{\partial}{\partial x}$	partial derivative
dx	differential of x
$y^{(k)}$	derivative of order k
$\overline{u}$	successive derivatives of u: $\overline{u} = (u, \dot{u}, \ddot{u}, \ldots)$
$\overline{u}^{(\alpha)}$	truncation of $\overline{u}$ at finite $\alpha \in \mathbb{N}$: $\overline{u} = (u, \dot{u}, \ldots, u^{(\alpha)})$
z	flat output
y^s	sensor measurement of y

y^z	redundant variable of y using the flat output z
μ	number of isolable faults
Σ	fault alarm signature
f_c	cutoff frequency

General Introduction

Context

Automatic control theory is a science that deals with the modeling, analysis, identification and control of dynamic systems. It is theoretically based on mathematics, signal theory and computer science. One of the main areas of automatic control is the planning of a reference trajectory that prepares the system to move from point A to point B. This involves the generation of reference inputs for the control system that ensure that the desired movement is achieved (Reif, 1979; Latombe, 1991; LaValle, 2006).

The use of the *flatness* property for nonlinear systems is one of the methods for constructing the reference trajectory. See surveys and books by (Lévine, 1999; Rudolph, 2003; Levine, 2009). Basically, a system is said to be flat if there is a vector called flat output, expressed from the input and state variables of the system, and such that all the variables of the system can be expressed as a function of this flat output and its successive time derivatives (Fliess et al., 1993). Therefore, after generating a reference trajectory of the flat output, the reference trajectories of the inputs, states and outputs can be deduced by simply deriving that of the flat output a finite number of times. For linear systems, the flatness property is equivalent to the controllability property, and the flat output is the variable resulting from the canonical form of Brunovský (Fliess et al., 1999).

The differential flatness property also exists for the class of fractional-order linear systems (Victor, 2010). These systems are modeled by fractional differential equations *e.g.* thermal systems (Battaglia et al., 2000), viscoelastic systems (Moreau, Ramus-Serment, and Oustaloup, 2002), nuclear magnetic resonance systems (Magin et al., 2008), *etc*. Like integer-order linear systems, a fractional linear system is flat if, and only if, it is controllable (Victor, 2010).

Another important area of automatic control is *fault tolerant control (FTC)* (Patton, 1993; Patton, 1997). For surveys see Gao, Cecati, and Ding (2015) and Ding (2012). Indeed, an automatic system is subject to malfunctions of its sensors, actuators or parameters. FTC is then introduced to design methods that allow the system to continue to operate safely when one or more of its components no longer function properly. An important step in fault tolerance monitoring is the *fault detection and isolation (FDI)* (Frank, 1990; Gertler, 1988). For surveys see Zhou, Xu, and Zhang (2014) and Thirumarimurugan, Bagyalakshmi, and Paarkavi (2016). Fault detection consists of detecting the abrupt change in system behavior, while fault isolation consists of determining the cause of the change.

Problem Statement

The flatness property of dynamic systems has also proven to be efficient in FDI. See Nan et al. (2008), Suryawan, De Doná, and Seron (2010) and Mai, Join, and Reger (2006). The latest flatness-based FDI method, introduced in Martínez-Torres et al. (2014), deals with additive and multiplicative faults on sensors and actuators. The residual signal, usually denoted by $r(t)$, is calculated by the difference between the measurement ζ^m of the sensor/actuator and its redundancy ζ^r. The redundant component, in its turn, is computed using the measurement of the flat output z, and its successive time derivatives, *i.e.*

$$\zeta^r = h_\zeta(z, \dot{z}, \ddot{z}, \ldots, z^{(\alpha)}).$$

Therefore, a necessary condition of this method to be applicable is that the measurement of the flat output vector z must be available at every time. Thus, it must be measured by sensors or at least deduced from the available measurements by flatness.

Practically, due to the presence of noise on system sensors and actuators, a threshold is fixed for each residue (Martínez-Torres et al., 2013a; Martínez-Torres et al., 2013b). Thus, if at least one residue exceeds its threshold then a fault on the system is detected, otherwise, no fault is detected. However, the isolability of faults

is more complex. The authors in Martínez-Torres et al. (2014) have shown that by using a single flat output vector, there may be faults that are detected but cannot be isolated.

A flat system admits an infinite number of flat outputs (Levine, 2009, Chapter 6), then, the authors in Martínez-Torres et al. (2014) proposed to use, if needed, several flat output vectors to increase the number of residues and that perhaps may ensure the full isolability of faults. However, we have noticed that the choice of the flat outputs is not arbitrary, and that they must be *independent* in the sense that when we use multiple flat outputs, we obtain a better fault isolability. Thus, the main problematic that this thesis addresses is the characterization of the flat outputs for the purpose of FDI.

Additionally, studies have shown that the classical FDI methods, basically developed to deal with integer-order systems, were not suitable for fractional-order systems, and hence fractional FDI methods were needed to be developed (Aoun et al., 2011). The second problematic of this thesis is to extend the characterization of the flat outputs to the class of fractional-order linear flat systems and apply this characterization to the FDI process of these systems.

This Thesis

This thesis is divided into two parts. The first one is dedicated to the characterization of the flat outputs for integer-order systems and the application on the FDI process. Chapter 1 presents a general overview on the FDI methods that exist in the literature, and provides attention to the methods based on the flatness property, while Chapter 2 presents a general overview on the appearance of the flatness property for the class of linear and nonlinear integer-order systems.

In Chapter 3, we recall two methods of computation of the flat outputs: the Smith decomposition and the unimodular completion algorithm. Then, a generalization of the flatness-based FDI method developed in Torres (2014) is presented, followed by a characterization of the flat outputs that are useful for the FDI. In Chapter 4, an application of the flat output characterization, proposed in Chapter 3, is performed on the classic example of hydraulic systems, the three-tank system.

The effectiveness of this characterization is demonstrated by simulations and experimentations.

The second part of this thesis is devoted to the extension of this characterization to the class of fractional-order linear flat systems. First in Chapter 5, we introduce a new algorithm of computation of the *fractional flat output*, based on the extension of the unimodular completion algorithm, developed in Fritzsche et al. (2016a), to the class of fractional-order linear flat systems. Then, in Chapter 6, we extend the flatness-based FDI method developed in Martínez-Torres et al. (2014) to the class of fractional-order linear systems. The effectiveness of this method is proved by simulations on the thermal bi-dimensional system.

Part I

Diagnostic of Integer-Order Flat Systems

Chapter 1

General Overview on Fault Detection and Isolation

Abstract

Fault tolerant control (FTC) refers to the ability of a system (computer, network, cloud cluster, etc.) to continue operating without interruption when one or more of its components fail. The goal of creating a fault-tolerant system is to prevent disruption from a single point of failure, ensuring high availability and business continuity of critical applications or systems. An important stage of fault tolerant control is the fault detection and isolation (FDI) of faults that appears on the system. In this chapter, a general overview of the different FDI methods that exist in the literature is presented, with particular attention to the methods based on the flatness property.

1.1 Fault tolerant control system

Fault tolerant control system (FTCS) refers to design control systems that allow a system to continue to operate without interruption, rather than fail completely, when one or more of its components fail. The FTCS approach must recover these following issues (Noura et al., 2009):

- Deals with different types of faults such as sensors, actuators and parameters;
- Detecting the fault and specifying its causes;
- Make a decision about the performance of the system.

FTCS can be classified into two types (Zhang and Jiang, 2008):

Passive FTCS: It consists of using a unique robust controller that deals with all the expected faults. This approach has limited fault tolerant capabilities, it is reliable only for the class of faults expected and taken into account in the design (Zhou, Doyle, Glover, et al., 1996).

Active FTCS: In contrast of the passive FTCS approach, here the controller is reconfigured online using the information from the FDI block, in order to maintain as best as possible the stability and some performance (Blanke et al., 2006; Noura et al., 2009).

The Active FTCS can be divided into four sub-systems (see Figure 1.1):

1. A reconfigurable feedforward/feedback controller, which reacts to the failure by designing a new controller.
2. A fault detection and isolation block. This block indicates whether or not there is a fault and specify its causes.
3. A controller reconfiguration mechanism connecting the fault identification mechanism and the reconfigurable controller.
4. A reference governor designed to generate a reference trajectory and to avoid actuator saturation.

Therefore, the FDI stage has an important role in the Active FTCS.

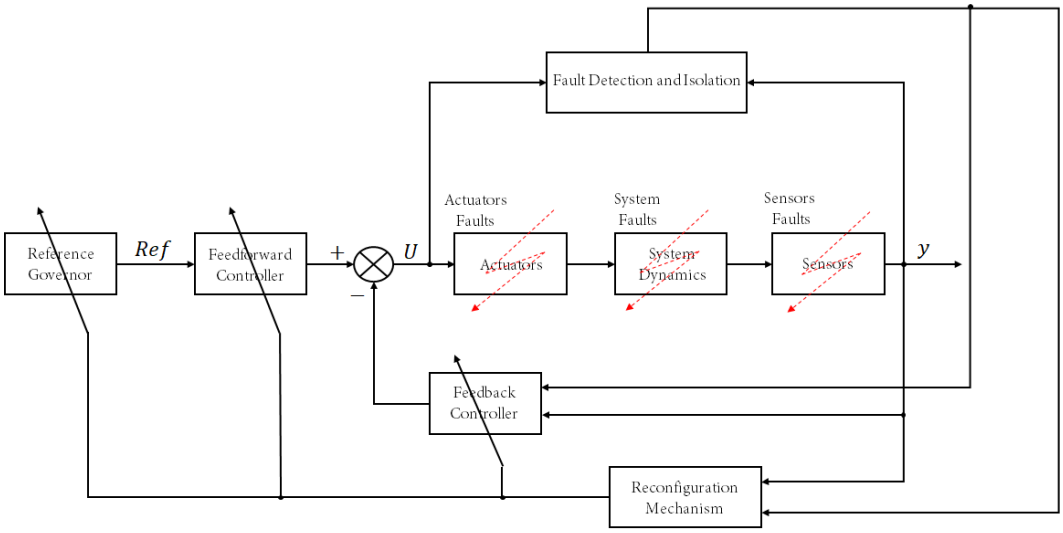

FIGURE 1.1: *A general structure of Active FTCS (Zhang and Jiang, 2008)*

1.2 Fault detection and isolation

The FDI problem has been introduced in automatic control as a paradigm for designing algorithms able to detect the outbreak of faults and identify their causes. Examples of survey papers on FDI may be found in Zhou, Xu, and Zhang (2014) and Thirumarimurugan, Bagyalakshmi, and Paarkavi (2016). The FDI process consists of two steps:

- **Fault detection:** consists in detecting the abrupt change in the behavior of the system;
- **Fault isolation:** consists in determining the exact location of the fault.

Over the last three decades, various FDI techniques have been developed. The first proposed method is the *hardware redundancy* in which redundant components are used in parallel to the process components (Chen et al., 2015). If the behavior of a process component is different from those of the redundant components then the fault is detected and isolated. The drawbacks of this method are the extra equipment, maintenance cost and additional space required to accommodate the equipment.

This approach was improved later on by the introduction of the *analytical redundancy* where the redundant component are calculated using algorithms which can

be implemented on some digital computer, and then avoid the extra equipment and the cost related to the hardware redundancy (Thirumarimurugan, Bagyalakshmi, and Paarkavi, 2016). A diagram showing the difference between the hardware redundancy and the analytical redundancy is illustrated in Figure 1.2.

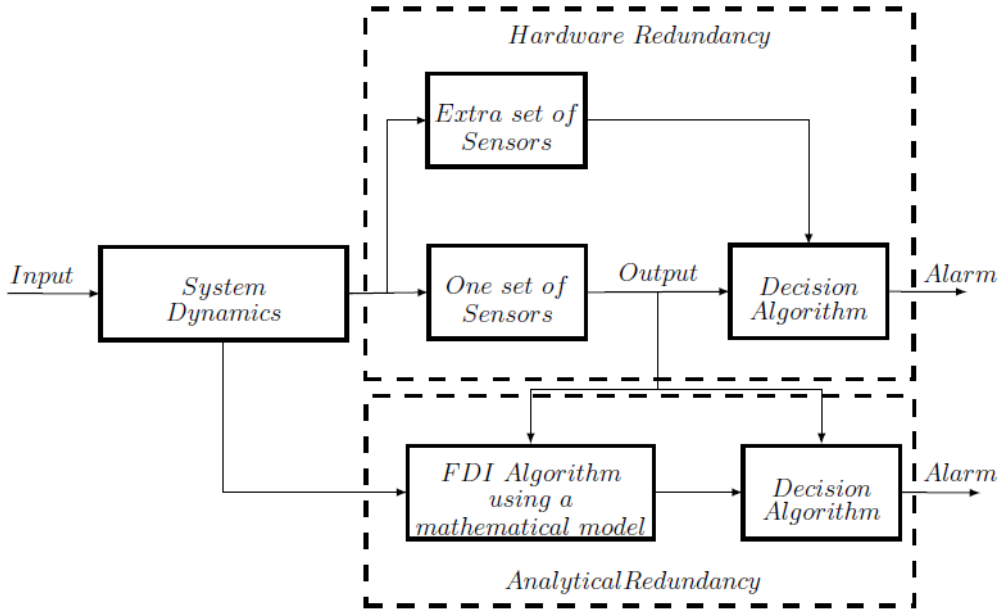

FIGURE 1.2: *Hardware redundancy and analytical redundancy diagram (Torres, 2014)*

The analytical redundancy methods are in general based on the notion of generating residual signals, or simply residues, usually denoted by $r(t)$, which are the difference between the measured output and the estimated process output. Then, the analytical redundancy FDI methods involve two stages: residual generation and residual evaluation, see Figure 1.3. In the ideal case where there is no uncertainties and disturbances on the system, the residue takes two values:

$$r(t) = 0 \qquad \text{in the fault-free case;}$$
$$r(t) \neq 0 \qquad \text{in the faulty case.}$$

In practice, this is not possible because it will always involve model uncertainties and device noise. Then the residue will be compared next to a threshold. If the residue exceeds its threshold then a fault is detected, otherwise there is no fault on the system. There exist several research studies on how to fix a threshold, the

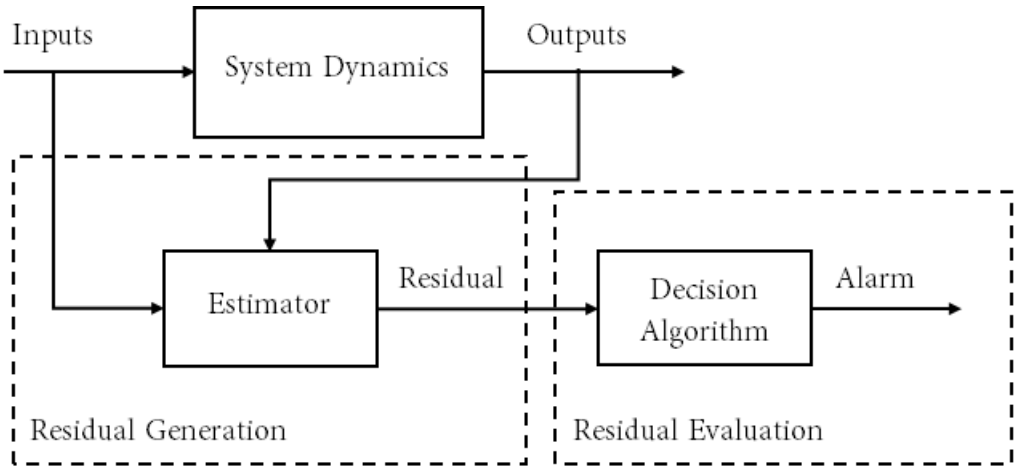

FIGURE 1.3: *Analytical redundancy fault detection and isolation*

interested reader may find them in Ding (2008), Khan and Ding (2011), Khan and Ding (2009), and Wu et al. (2017).

The analytical redundancy FDI methods can be categorized into three main classes: model-based methods, data driven methods and hybrid methods (Zhou, Xu, and Zhang, 2014).

Model-based methods: They are based on the knowledge of the mathematical model of the system. Here, the residues are constructed using mathematical equations describing the plant. Examples of model-based methods are the observer-based approach (Chen and Patton, 2012; Ding, 2008; Frank, Schrier, and Garcia, 1999), parity-space approach Chow and Willsky, 1984; Gertler and Singer, 1990 and parameter estimation approach (Crassidis and Junkins, 2011; NøRgaard, Poulsen, and Ravn, 2000).

Data driven methods: They are based on the knowledge of a wide part of the history of the process (Venkatasubramanian et al., 2003). Examples of data driven approaches are the Fuzzy logic (Dexter and Ngo, 2001) and artificial neural network (ANN) (Frank and Ding, 1997).

Hybrid methods: They are the combination between the model-based methods and the data driven methods (Patan and Parisini, 2005; Raie and Rashtchi, 2002; Zhou, Liu, and Dexter, 2014).

1.3 Flatness-based FDI

Recently, the flatness property has been introduced into the repertoire of FDI techniques. It falls under the model-based analytical redundancy methods. Roughly speaking, a dynamic system of the form

$$\begin{cases} \dot{x} = f(x,u) \\ y = h(x,u) \end{cases} \tag{1.1}$$

with $x = (x_1, \ldots, x_n)$ is the n-dimensional state vector, $u = (u_1, \ldots, u_m)$ is the input vector of dimension m and $y = (y_1, \ldots, y_p)$ is the output vector of dimension p, is said to be flat if, and only if, there exists a vector $z = (z_1, \ldots, z_m)$ of dimension m, called flat output, such that:

$$z = \psi(x, u, \dot{u}, \ldots, u^{(\alpha)}) \tag{1.2}$$

and conversely,

$$\begin{aligned} x &= \varphi_x(z, \dot{z}, \ldots, z^{(\beta)}) \\ u &= \varphi_u(z, \dot{z}, \ldots, z^{(\beta+1)}) \\ y &= h(\varphi_x(z, \dot{z}, \ldots, z^{(\beta)}), \varphi_u(z, \dot{z}, \ldots, z^{(\beta+1)})). \end{aligned} \tag{1.3}$$

An example of the flatness-based FDI methods is the use of a nonlinear observer coupled with the differential flatness (Nan et al., 2008). In this method, the input variables are estimated in both ways: using the observer and using the flat output, and then these two redundancies are compared to generate the residues. The drawback of this method is the use of two different approaches which could differ in some aspects, for example in the dynamic speed and could create false alarms.

In Mai, Join, and Reger (2006) and Mai, Join, and Reger (2007), the flatness-based FDI method uses an algebraic approach to estimate actuator faults. It takes into account only additive faults.

In Suryawan, De Doná, and Seron (2010), the flat output is used to calculate the redundant variables. Since this property evaluates the derivatives of the flat

output and, due to the presence of noise, these derivatives may not exist, then they have to be estimated. In Suryawan, De Doná, and Seron (2010), the flat output successive derivatives are estimated using B-splines. This approach is applied to linear systems and sensor faults only. In addition, the estimation process could take time to be accomplished.

Also, due to the presence of noise, a probabilistic distribution is generated in López-Fernandez and Olivella (2009) and Zhang et al. (2012), in order to model errors and improve the effectiveness of the threshold-based fault detection scheme. The drawback of this method is the fact that creating an online probabilistic distribution is difficult.

In Torres (2014), a novel flatness-based FDI method has been developed. The redundant variables are calculated using a flat output. The derivatives of the flat output are estimated using a high-gain observer coupled with a low-pass filter to improve its performance (Martínez-Torres et al., 2014). In this method, the full state vector x of the system (1.1) is supposed to be equal to the output y, *i.e.* x is measured by sensors, and the input signals are supposed to be available at every time. Moreover, a necessary condition of this method to be applicable is that the measurements of the flat output components are available too. Additionally, in Martínez-Torres et al. (2014), the components of the flat output are supposed to be part of the measured state x, *i.e.* $z = \mathrm{pr}_{\mathbb{R}^m}(x)$. Therefore, the redundant state $\widehat{x}_k$, for $k = 1, \ldots, n$ and input variables $\widehat{u}_l$ for $l = 1, \ldots, m$, are constructed using (1.3) as follows:

$$\begin{aligned} \widehat{x}_k &= \varphi_{0,k}(z, \dot{z}, \ldots, z^{(\beta)}) \\ \widehat{u}_l &= \varphi_{1,l}(z, \dot{z}, \ldots, z^{(\beta+1)}). \end{aligned} \tag{1.4}$$

Then, the residual signal r_k, associated to the sensor S_k, is given by:

$$r_k = x_{k_m} - \widehat{x}_k \tag{1.5}$$

where x_{k_m} is the real measurement of the sensor S_k. A residual signal associated to an actuator A_l is defined in the same way:

$$r_l = u_{l_m} - \widehat{u}_l. \tag{1.6}$$

This method deals with additive and multiplicative faults on both sensors and actuators. It is applicable on both linear and nonlinear flat systems. A scheme of this method is illustrated in Figure 1.4.

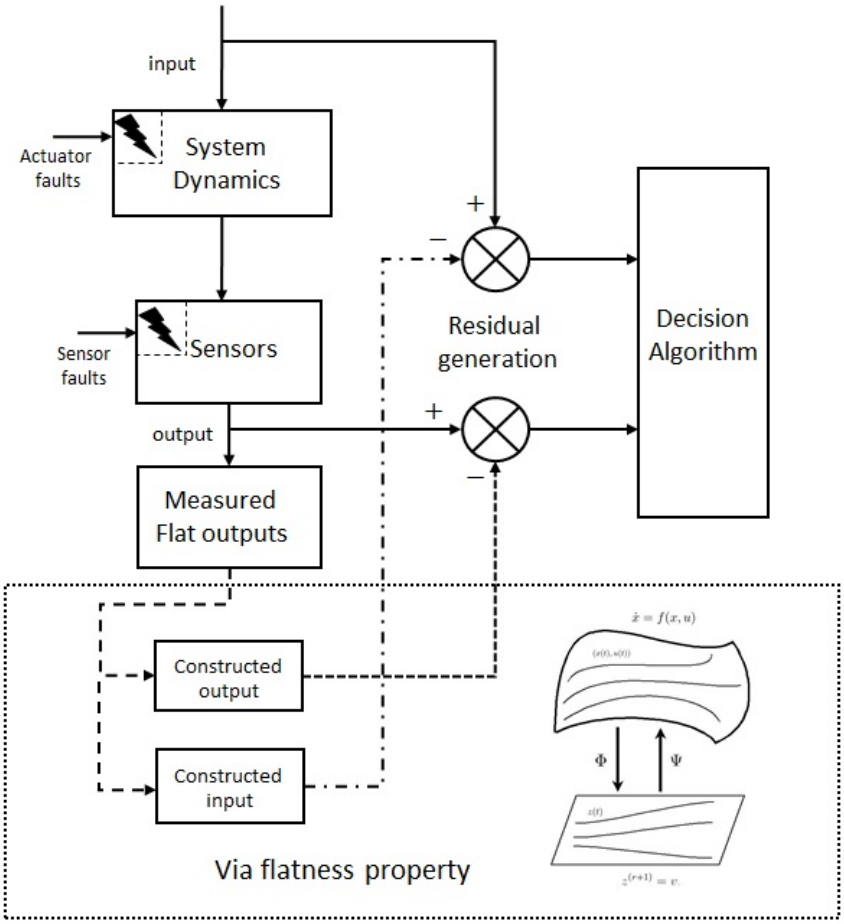

FIGURE 1.4: *Flatness-based FDI*

The fault detection is simple and common in all the above methods: if a residue exceeds its threshold, then a fault alarm is generated. However, the isolability is more complex. It depends on the chosen flat output, and sometimes we need multiple flat outputs to isolate all faults (Martínez-Torres et al., 2013a). However, the choice of these flat outputs is not arbitrary, *i.e.* there are flat outputs that, when used

together, increase the isolability of faults and others that do not. Thus, the used flat outputs must be independent in the sense that, by using them together, we obtain a better fault isolability.

1.4 Conclusion

The FTC process is an important issue in automatic control. It consists of preventing system disruptions from a single point of failure. An important stage of FTC process is the FDI. It consists of designing algorithm able to detect faults and isolate their causes. In this chapter, a brief overview on the FDI methods has been presented with a particular attention to the methods based on the flatness property.

The latest flatness-based FDI method has been developed by Martínez-Torres et al. (2014). In this method, the flat outputs are used to construct the vector of residues. The isolability of faults, using this method, sometimes requires multiple flat outputs. However, the choice of these flat outputs is not arbitrary. In Chapter 3 of this thesis, a characterization of the flat outputs is introduced, which leads to a decision algorithm on the choice of flat outputs that are useful for the FDI process. Before that, in the next chapter, a general overview on the flatness property of linear and nonlinear systems is presented.

Chapter 2

Differential Flatness for Integer-Order Nonlinear Systems

Abstract

Controllability is an important property in control system theory. It plays a crucial role in many control problems, such as stabilization of unstable systems by feedback, or optimal control. The concept of controllability refers to the ability to move a system throughout its configuration space using only certain permissible manipulations. An important application of the controllability is the transformation of the system into a simpler form, called controllability canonical form, and hence facilitate the planning of a reference trajectory and the tracking of this trajectory. The controllability property has been first studied for the class of linear systems, and then extended in many different ways to the class of nonlinear systems. The differential flatness property is one of these extensions. In this chapter, recalls on the controllability property of linear systems and controllability canonical form are presented, as well as a general overview on the flatness property of nonlinear systems. An illustration of the application of the flatness property to trajectory planning is also provided.

2.1 Introduction

The controllability property of dynamic systems, known as Kalman's controllability, has been first introduced for the class of linear systems of the form:

$$\dot{x} = Ax + Bu, \tag{2.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector and $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ are real matrices. Moreover, it has been shown that every linear controllable system, *i.e.* a linear system that verifies the controllabilty property, can be transformed, by change of coordinates, into a simpler system called Brunovskỳ's canonical form (see Levine, 2009, Chapter 4).

Likewise, the *differential flatness* is a property of nonlinear dynamic systems that allows the transformation of a highly complex system of differential equations of the form:

$$\dot{x} = f(x, u) \tag{2.2}$$

into a simpler one called *flat system*. As mentioned in Fliess et al. (1995b), "*The flatness might be seen as another nonlinear extension of Kalman's controllability*". In other words, a linear controllable system is a flat system (Fliess et al., 1999).

The differential flatness property has been introduced by Martin and colleagues, in Fliess et al. (1992) and Martin (1992), to deal with the class of nonlinear systems (2.2). Roughly speaking, a system is said to be flat if, and only if, all the state and input variables can be expressed in function of a new variable z, called *flat output*, and its successive time derivatives:

$$x = \Phi_x(z, \dot{z}, \ddot{z}, \ldots) \tag{2.3}$$

$$u = \Phi_u(z, \dot{z}, \ddot{z}, \ldots) \tag{2.4}$$

and in turn, z is function of x, u and successive time derivatives of u.

Numerous successful applications have been created, reflecting the importance of the differential flatness property in many areas such as robotics (Kiss, Levine, and Lantos, 1999), non-holonomic vehicles (Fliess et al., 1995a), aeronautic (Martin,

Devasia, and Paden, 1996), electrical motors (Marquez and Delaleau, 1999), hydraulic (Bindel et al., 2000), chemical engineering (Petit, 2000; Rothfuss, Rudolph, and Zeitz, 1996), magnetic bearings (Levine, Lottin, and Ponsart, 1996) and automobile industry (Bitauld, Fliess, and Lévine, 1997; Lévine and Rémond, 2000).

The flatness property is well adapted to trajectory planning (Yang, Pan, and Wan, 2019), trajectory tracking (Antritter, Müller, and Deutscher, 2004) and designing robust controllers (Cazaurang, 1997; Lavigne, 2003).

In Section 2.2 of this chapter, we recall the notions of system controllability and controllability canonical forms for linear systems. Then, we recall their extensions to nonlinear systems in Section 2.3, which leads to the definition of the flatness property, described in Section 2.4. Finally, Section 2.5 is devoted to the application of the flatness property to the generation of reference trajectories.

2.2 Controllability of linear systems

In control theory, the system *controllability* and the *controllability canonical form* represent a first approach to solve the problem of path planning and path tracking. Path planning consists of building offline trajectories, called reference trajectories, with associated inputs, based on knowledge of the model and in the absence of disturbances. These trajectories link an initial state to a final state in an open-loop system. However, path tracking is based on the design of a control law, allowing to follow a reference trajectory. The controllability property itself is roughly defined as an ability to do whatever possible with a system. In more technical terms, the ability to transfer the system from an initial state $x(t_0) = x_0$ to a final state $x(t_T) = x_T$ in a finite time $T < \infty$. Then, a system is controllable if, and only if, we can find a control law $u(t)$ that brings x_0 to x_T. In addition, the controllability of a system provides it with the flexibility to be transformed, by a change of coordinates, to a simpler form called controllability canonical form or Brunovský's canonical form. This transformation is not verified for all controllable systems and the system must verify the flatness property, as shown later.

The controllability property has initially been studied by Kalman et al. (1960) for the class of linear systems and then extended in several ways to the class of

nonlinear systems. In this section, we recall the controllability property for the class of linear systems and the related controllability canonical form.

2.2.1 System controllability

Consider the following linear system:

$$\dot{x} = Ax + Bu \tag{2.5}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector and A and B are real matrices of size $n \times n$ and $n \times m$, respectively.

Definition 2.2.1 (Kalman, 1963) *We say that the system* (2.5) *is* controllable *if, and only if, for any given duration $T > 0$ and two points x_0 and x_T of $\mathbb{R}^n$, there exists a control law $t \in [0, T] \mapsto u(t) \in \mathbb{R}^m$, piece-wise continuous, that brings the system from state x_0 to state x_T, i.e. the solution $x(t)$ of* (2.5)*, generated by u, satisfies $x(T) = x_T$. In other words:*

$$e^{AT}x_0 + \int_0^T e^{A(T-t)}Bu(t)dt = x_T. \tag{2.6}$$

In fact, this property depends only on the pair (A, B) as presented in the following theorem:

Theorem 2.2.1 (Kalman criterion) *The system* (2.5) *is controllable if, and only if, the following matrix*

$$\mathcal{E} = (B \vdots AB \vdots \cdots \vdots A^{n-1}B) \tag{2.7}$$

is of rank n.

The proof of this theorem is detailed in Kailath (1980).

2.2.2 Controllability canonical form

According to Brunovskỳ (1970), for every linear controllable system, one can find a change of coordinates such that the system (2.5) is equivalent to the canonical form:

$$z_i^{(n_i)} = v_i, \quad i = 1, \ldots, m \quad \text{and} \quad \sum_{i=1}^{m} n_i = n, \tag{2.8}$$

where z_i and v_i are the new state and input variables, respectively, and the indices n_i, for $i = 1, \ldots, m$, are called *controllablility indices* or *Brunovský indices*. We also say that the two systems (2.5) and (2.8) are *static linear feedback equivalent* or simply *SLF-equivalent* according to the following definition:

Definition 2.2.2 (SLF-equivalence) *Two linear systems $\dot{x} = Ax + Bu$ and $\dot{z} = Fz + Gv$ are said SLF-equivalent or equivalent by linear change of coordinates if, and only if, there exist two invertible matrices M and L, of size $n \times n$ and $m \times m$ respectively, and a matrix K of size $m \times n$, such that if x and u satisfy $\dot{x} = Ax + Bu$ and $z = Mx$ and $v = Kx + Lu$ then z and v satisfy $\dot{z} = Fz + Gv$, and conversely.*

Indeed, for the simple case of $m = 1$, (2.5) becomes

$$\dot{x} = Ax + bu \tag{2.9}$$

with $b \in \mathbb{R}$, the controllability condition (2.7) implies that the vectors $\{b, Ab, \ldots, A^{n-1}b\}$ form a basis of the space $\mathbb{R}^n$. Then $\{b, Ab, \ldots, A^{n-1}b\}$ is isomorphic to the canonical base $e = (e_1, \ldots, e_n)$[1] of $\mathbb{R}^n$. This canonical base is itself generated by the vector g and the matrix F given by:

$$g = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \quad F = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & & \ddots & \ddots & \\ 0 & 0 & 0 & & 1 \\ 0 & 0 & 0 & \ldots & 0 \end{pmatrix}. \tag{2.10}$$

[1]The vector e_i has 1 on the i$^{\text{th}}$ position and 0 elsewhere.

One can easily verify that the system (2.9) is SLF-equivalent to the system $\dot{z} = Fz + gv$, which, according to (2.10), is given by:

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \vdots \\ \dot{z}_{n-1} = z_n \\ \dot{z}_n = v \end{cases} \iff z_1^{(n)} = v. \tag{2.11}$$

For the general case where $m > 1$, the canonical form of the system (2.5) is constructed by blocks, each block has the vector g and the matrix F given by (2.10):

Theorem 2.2.2 (Brunovskỳ, 1970) *Every linear controllable system given by the pair (A, B) is SLF-equivalent to its canonical form defined by $F = diag\{F_1, \ldots, F_m\}$, $G = diag\{g_1, \ldots g_m\}$, where each pair (F_i, g_i) is given by* (2.10)*, $i = 1, \ldots, m$, with F_i of size $n_i \times n_i$ and g_i of size $n_i \times 1$, the integers $n_1, \ldots, n_m$ being the controllability indices of (A, B) and satisfying $1 \leq n_i \leq n$ and $\sum_{i=1}^{m} n_i = n$.*

In other words, every linear controllable system $\dot{x} = Ax + Bu$ is SLF-equivalent to the system $z_i^{(n_i)} = v_i$ for $i = 1, \ldots, m$ and $\sum_{i=1}^{m} n_i = n$.

According to Theorem 2.2.2 and Definition 2.2.2 one can easily see that

$$x = M^{-1}z \quad \text{and} \quad u = -L^{-1}KM^{-1}z + L^{-1}v, \tag{2.12}$$

then, there exists a linear invertible map ϕ such that:

$$x = \phi(z_1, \ldots, z_1^{(n_1-1)}, \ldots, z_m, \ldots, z_m^{(n_m-1)}) \tag{2.13}$$

and a map ψ such that:

$$u = \psi(z_1, \ldots, z_1^{(n_1)}, \ldots, z_m, \ldots, z_m^{(n_m)}) \tag{2.14}$$

where the partial function $(z_1^{(n_1)}, \ldots, z_m^{(n_m)}) \mapsto (u_1, \ldots, u_m)$ is linear invertible for all

$(z_1, \dots, z_1^{(n_1-1)}, \dots, z_m, \dots, z_m^{(n_m-1)})$. Since the matrix M^{-1} is invertible, the application ϕ is a diffeomorphism[2].

2.3 Controllability of nonlinear systems

For nonlinear systems of the form

$$\dot{x} = f(x, u), \tag{2.15}$$

several notions of controllability may be defined and can be found in literature, notably local controllability around an equilibrium point, first-order controllability, strong controllability and weak controllability. The local controllability around an equilibrium point and the first-order controllability are directly inspired by the previous linear analysis.

Let $(\widetilde{x}, \widetilde{u})$ be an equilibrium of the system, *i.e.* $f(\widetilde{x}, \widetilde{u}) = 0$, and

$$\dot{x} = Ax + Bu, \quad \text{with} \quad A = \frac{\partial f}{\partial x}(\widetilde{x}, \widetilde{u}) \quad \text{and} \quad B = \frac{\partial f}{\partial u}(\widetilde{x}, \widetilde{u}) \tag{2.16}$$

be the tangent linear system, associated with (2.15), around the equilibrium point $(\widetilde{x}, \widetilde{u})$. The next two definitions are borrowed from Levine (2009):

Definition 2.3.1 (Local controllability) *The system* (2.15) *is locally controllable at the equilibrium point* $(\widetilde{x}, \widetilde{u})$ *if for all real* $\epsilon > 0$ *there exists a real* $\eta > 0$ *such that for every pair of points* $(x_0, x_1) \in \mathbb{R}^n \times \mathbb{R}^n$ *satisfying* $\|x_0 - \widetilde{x}\| < \eta$ *and* $\|x_1 - \widetilde{x}\| < \eta$, *there exists a piece-wise continuous control* u *on* $[0, \epsilon]$ *such that* $\|u(t)\| < \epsilon \ \forall t \in [0, \epsilon]$ *and* $X_\epsilon(x_0, u) = x_1$, *where* $X_\epsilon(x_0, u)$ *is the integral curve* [3] *of* (2.15) *at time* ϵ, *generated by* u *from* x_0.

Definition 2.3.2 (First-order controllable) *We say that the system* (2.15) *is first-order controllable at the equilibrium point* $(\widetilde{x}, \widetilde{u})$ *if the rank of the matrix* $\mathcal{E}$ *defined by* (2.7) *for the tangent linear system* (2.16), *is equal to* n.

[2] A differentiable map f is said to be diffeomorphism if it is invertible and its inverse f^{-1} is also differentiable.

[3] Trajectories that are solutions of the system of differential equations (2.15).

The relation between these two definitions of controllability of nonlinear systems is summarized in the following theorem:

Theorem 2.3.1 (Levine, 2009) *Every first-order controllable system* (2.15) *at an equilibrium point* $(\widetilde{x}, \widetilde{u})$ *is locally controllable. However, a nonlinear system* (2.15) *can be locally controllable without being first-order controllable.*

As presented in Section 2.2.2, every linear controllable system can be transformed by a change of coordinates to the canonical form (2.8), called Brunovskỳ's canonical form. This property has been generalized for the class of nonlinear systems, with the understanding that the application ϕ of equation (2.13) loses the property of global diffeomorphism between the two systems. This generalization to the class of nonlinear systems leads to the introduction of the differential flatness property and the so-called *Lie-Bäcklund isomorphism*.

Lie-Bäcklund isomorphism

Starting with an example, consider a four-wheeled vehicle running without sliding on the horizontal plane $(O, \overrightarrow{i}, \overrightarrow{j})$, see Figure 2.1. We denote by $(x, y)^T$ the coordinates of the point P, the middle of the rear axle, Q the middle of the front axle, θ the angle between the longitudinal axis of the vehicle and the (Ox) axis and φ the angle of the front wheels. The vehicle speed module is denoted by $u = \left\| \frac{d\overrightarrow{OP}}{dt} \right\| \cdot \frac{\overrightarrow{PQ}}{\|\overrightarrow{PQ}\|}$, where $\|\overrightarrow{PQ}\| = l$. An elementary kinetic calculation gives the following explicit system of equations:

$$\begin{cases} \dot{x} = u \cos\theta \\ \dot{y} = u \sin\theta \\ \dot{\theta} = \dfrac{u}{l} \tan\varphi \end{cases} . \tag{2.17}$$

Clearly, system (2.17) is a nonlinear system, with $(x, y, \theta) \in X \subset \mathbb{R}^3$ the state vector and $(u, \varphi) \in U \subset \mathbb{R}^2$ the input vector. From the first two equations we can see that:

$$\theta = \tan^{-1}\left(\frac{\dot{y}}{\dot{x}}\right) \quad \text{and} \quad u = \sqrt{\dot{x}^2 + \dot{y}^2}. \tag{2.18}$$

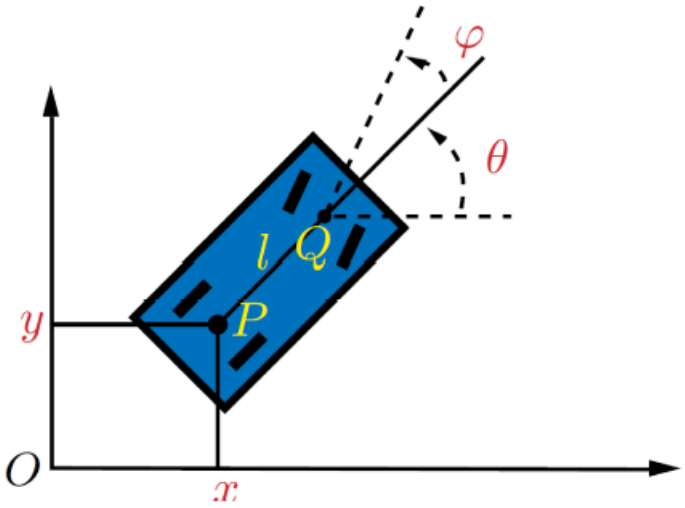

FIGURE 2.1: *Non-holonomic vehicle, Source: Levine (2009)*

Deriving the expression of θ in (2.18), we obtain:

$$\dot{\theta} = \frac{\ddot{y}\dot{x} - \dot{y}\ddot{x}}{\dot{x}^2 + \dot{y}^2}. \tag{2.19}$$

and then we get:

$$\varphi = \tan^{-1}\left(l\frac{\ddot{y}\dot{x} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}\right) \tag{2.20}$$

Then, all the trajectories of the system can be described through the variables x and y and their successive derivatives up to the order 2. By setting:

$$\begin{aligned} z_1^{(2)} &= x^{(2)} = v_1 \\ z_2^{(2)} &= y^{(2)} = v_2, \end{aligned} \tag{2.21}$$

we obtain a form similar to the linear canonical form (2.8). Consequently, there exists a transformation Φ such that for every $(x, \dot{x}, v_1, y, \dot{y}, v_2)$ there exists $(x, y, \theta, u, \varphi)$ that satisfies the above-mentioned equations. Moreover, the transformation Φ is invertible in the sense that it maps trajectories to trajectories in a one-to-one way. That is, for every trajectory $t \mapsto (z_1(t), z_2(t))$, there exists a unique local trajectory $t \mapsto (x(t), y(t), \theta(t), u(t), \varphi(t))$, and conversely by $z_1(t) = x(t)$ and $z_2(t) = y(t)$. In

addition, the transformation Φ is compatible with the operator of differentiation $\frac{d}{dt}$: by denoting $\chi = (x, y, \theta, u, \varphi)$, the system (2.17) becomes $F(\chi, \dot{\chi}) = 0$ and we have:

$$F(\Phi, \dot{\Phi}) = 0. \tag{2.22}$$

However, the initial system (2.17) is of dimension 3 (the system belongs to a manifold of dimension 3), whereas its equivalent canonical form (2.21) is of dimension 4:

$$\sum_{i=1}^{2} n_i = 4. \tag{2.23}$$

Then, according to the constant rank theorem (see Levine, 2009, Chapter 2), Φ is not a diffeomorphism because it does not preserve the dimension of the system.

This issue in the case of nonlinear systems can be excluded by extending the mapping Φ from $(x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y})$ to $(x, y, \theta, u, \varphi)$ to the mapping $\overline{\Phi} = (\Phi, \dot{\Phi}, \ddot{\Phi}, \ldots)$ from $(x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y}, \ldots)$ to $(x, y, \theta, u, \varphi, \dot{x}, \dot{y}, \dot{\theta}, \dot{u}, \dot{\varphi}, \ldots)$. In this case, $\overline{\Phi}$ maps a space of infinite dimension to a space of the "same" infinite dimension. In this context, $\overline{\Phi}$ is called Lie-Bäcklund isomorphism and it admits the following characteristics (Levine, 2009, Chapter 5):

- every component of the transformation is function of the coordinates and their successive derivatives up to an order not known in advance;
- this transformation is invertible in the sense that we can go back to the original coordinates by a transformation of the same type;
- it is compatible with the operator of the differentiation $\frac{d}{dt}$.

The canonical form (2.21) for the class of nonlinear systems is called flat system, and the vector (z_1, z_2) is called flat output of the system. Definitions of the flatness, flat systems and flat outputs are detailed in the next section.

2.4 Differential flatness

Based on the previous sections, the flatness property has been introduced to deal with the controllability of nonlinear systems. Moreover, a nonlinear system that can be transformed by a change of coordinates, called flat outputs, to its Brunovský's canonical form, is a flat system. In this section, we recall the definitions of the flatness property for nonlinear and linear systems.

2.4.1 Nonlinear flat system

Consider the following nonlinear system:

$$\dot{x} = f(x, u) \tag{2.24}$$

where $x = (x_1, \ldots, x_n)^T$, the state vector, evolves in a n-dimensional manifold X, $u \in \mathbb{R}^m$ is the input vector, f a $\mathcal{C}^\infty$ function[4] of x and u with $rank\left(\frac{\partial f}{\partial u}\right) = m$ and $m \leq n$. For the reasons explained in the previous section, we extend the manifold $X \times \mathbb{R}^m$ to the manifold of jets of infinite order (Levine, 2009, Chapter 5):

$$\mathfrak{X} \triangleq X \times \mathbb{R}^m \times \mathbb{R}^m \times \ldots = X \times \mathbb{R}^m_\infty. \tag{2.25}$$

Coordinates in the manifold $\mathfrak{X}$ are of the form:

$$(x, \overline{u}) \triangleq (x, u, \dot{u}, \ddot{u}, \ldots). \tag{2.26}$$

By applying the *implicit function theorem* (see Levine (2009, Chapter 2)), we get the following implicit system, associated to (2.24):

$$F(x, \dot{x}) = 0 \tag{2.27}$$

[4]The function f is said to be infinitely differentiable, smooth, or of class $\mathcal{C}^\infty$, if it has derivatives of all orders.

where $F \in \mathcal{C}^\infty$ from TX, the tangent space of X [5], to $\mathbb{R}^{(n-m)}$ such that *rank* $\left(\frac{\partial F}{\partial \dot{x}}\right) = n - m$. The corresponding manifold of jets of infinite order is $\mathfrak{X} = X \times \mathbb{R}^n_\infty$ with coordinates:

$$\overline{x} \triangleq (x, \dot{x}, \ddot{x}, \ldots). \tag{2.28}$$

Integral curves of the systems (2.24) and (2.27) coincide on the manifold $\mathfrak{X}_0$ defined by

$$\mathfrak{X}_0 = \{\overline{x} \in \mathfrak{X} \mid \frac{d^k}{dt^k} F(x, \dot{x}) = 0, \forall k \in \mathbb{N}\} \setminus \{\overline{x} \in \mathfrak{X} \mid \nexists\, u \in \mathbb{R}^m \text{ s.t. } \dot{x} - f(x, u) = 0\}. \tag{2.29}$$

(see Kaminski, Lévine, and Ollivier (2018)).

In the sequel, we systematically denote by $\overline{\xi} \triangleq (\xi, \dot{\xi}, \ddot{\xi}, \ldots)$ the sequence of infinite order jets of a vector ξ, and $\overline{\xi}^{(\alpha)} \triangleq (\xi, \dot{\xi}, \ldots, \xi^{(\alpha)})$ the truncation at the finite order $\alpha \in \mathbb{N}$ of the previous sequence.

Definition 2.4.1 (Levine, 2009) *The system* (2.24) *is called differentially flat at a point* $(x_0, \overline{u}_0) \in \mathfrak{X}_0$ *if, and only if, there exist a vector* $z = (z_1, \ldots, z_m) \in \mathbb{R}^m$, *two integers* ρ *and* v *and a Lie-Bäcklund isomorphism* ψ *defined on a neighbourhood* $\mathcal{V}$ *of* $(x_0, \overline{u}_0)$ *in* $\mathfrak{X}_0$ *with inverse* $\varphi = (\varphi_0, \varphi_1, \ldots)$ *defined on a neighbourhood* $\mathcal{W} \subset \psi(\mathcal{V})$ *of* $\overline{z} \triangleq (z, \dot{z}, \ddot{z}, \ldots) \triangleq \psi(x_0, \overline{u}_0)$ *in* $\mathbb{R}^m_\infty$ *such that:*

1. $z = \psi(x, u, \dot{u}, \ldots, u^{(v)}) \in \mathcal{W}$;
2. $z_1, \ldots, z_m$ *and their successive derivatives are linearly independent in* $\mathcal{W}$;
3. *The state* x *and the input* u *are functions of* z *and its successive derivatives:*

$$(x, u) = (\varphi_0(\overline{z}^{(\rho)}), \varphi_1(\overline{z}^{(\rho+1)})) \in \mathrm{pr}_{X \times \mathbb{R}^m}(\mathcal{V}) \tag{2.30}$$

where $\mathrm{pr}_{X \times \mathbb{R}^m}(\mathcal{V})$ *is the canonical projection from* $\mathcal{V}$ *to* $X \times \mathbb{R}^m$;

4. *The differential equation* $\dot{\varphi}_0(\overline{z}) = f(\varphi_0(\overline{z}), \varphi_1(\overline{z}))$ *is identically satisfied in* $\mathcal{W}$.

[5] The tangent space at a point x of a differential manifold X is a vector space which is the set of all the possible velocity vectors of a body moving in the variety X when it is in x: $TX = \cup_{x \in X} T_x X = \cup_{x \in X} \{(x, \dot{x}) \mid x \in X)\}$.

The vector z is called *flat output* of the system. The system (2.24) can be written into the canonical form:

$$\begin{cases} z_1^{(\rho_1+1)} = v_1 \\ \quad \vdots \\ z_m^{(\rho_m+1)} = v_m \end{cases} \tag{2.31}$$

where $v = (v_1, \ldots, v_m)^T$ is the new input variable and $\rho = (\rho_1, \ldots, \rho_m)^T$.

Example 2.4.1 (Non-holonomic vehicle (Levine, 2009)) *In the non-holonomic vehicle example presented in section 2.3, all the state and input variables of the system* (2.17) *are functions of x, y and their successive derivatives. Then the system* (2.17) *is a flat system with $z = (x, y)^T$ the flat output.*

Example 2.4.2 (Inverted pendulum (Levine, 2009)) *The dynamic of the inverted pendulum, located in the plane (x, z), is represented by the following explicit system:*

$$\begin{cases} m\ddot{x}_c = F_x \\ m\ddot{z}_c = F_z - mg \\ \dfrac{J}{d}\ddot{\theta} = F_z \sin\theta - F_x \cos\theta \end{cases} \tag{2.32}$$

where m is the mass of the pendulum, (x_c, z_c) is the position of the center of mass of the pendulum in the plane (x, z), θ is the angle between the pendulum and the vertical and J is the inertia of the pendulum, see Figure 2.2. This system is controlled by an exterior force

$$\overrightarrow{F} = F_x \overrightarrow{i} + F_z \overrightarrow{k} \tag{2.33}$$

applied on a point A located at a distance d of the center of mass C.

Denoting by:

$$x_1 = \frac{x_c}{g}, \quad x_2 = \frac{z_c}{g}, \quad x_3 = \theta, \quad u_1 = \frac{F_x}{mg}, \quad u_2 = \frac{F_z}{mg} - 1, \quad \varepsilon = \frac{J}{mgd}$$

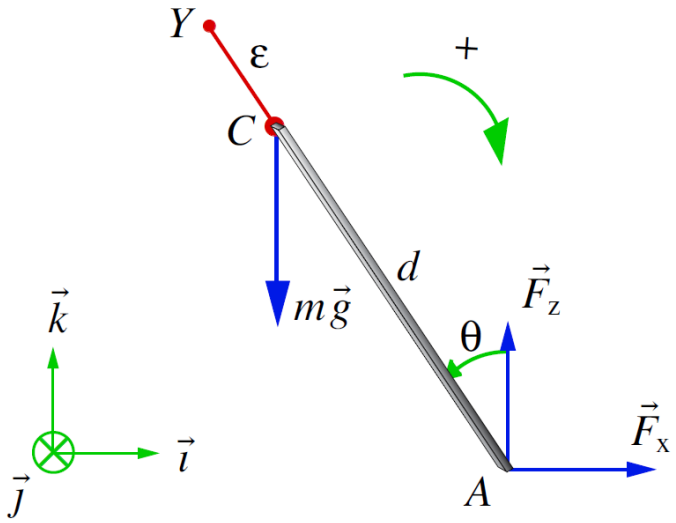

FIGURE 2.2: *Inverted pendulum, Source: Levine (2009)*

the system (2.32) *becomes:*

$$\begin{cases} \ddot{x}_1 = u_1 \\ \ddot{x}_2 = u_2 \\ \varepsilon \ddot{x}_3 = -u_1 \cos x_3 + (u_2 + 1) \sin x_3 \end{cases} . \tag{2.34}$$

The state vector is then $x = (x_1, x_2, x_3)^T \in \mathbb{R}^3$ *and the input vector is* $u = (u_1, u_2)^T \in \mathbb{R}^2$. *Let* (ξ, ζ) *be the coordinates of the point* Y. *It satisfies*

$$(\xi - x)^2 + (\zeta - z)^2 = \varepsilon^2 \tag{2.35}$$

and the difference between its acceleration and the normalized acceleration of gravity $-\overrightarrow{k}$, *is co-linear to the vector* $\overrightarrow{YC}$:

$$\frac{\ddot{\xi}}{\ddot{\zeta} + 1} = \frac{\xi - x_1}{\zeta - x_2} \tag{2.36}$$

In polar coordinates, we can write:

$$\xi = x_1 + \varepsilon \sin x_3, \quad \zeta = x_2 + \varepsilon \cos x_3 \tag{2.37}$$

and Y *is called the oscillation center or the Huygens oscillation center (Fliess et al., 1999). The system* (2.34) *is flat with* (ξ, ζ) *a flat output. In fact, from equations* (2.36) *and* (2.37)

we have

$$x_3 = \tan^{-1}\left(\frac{\xi - x_1}{\zeta - x_2}\right) = \tan^{-1}\left(\frac{\ddot{\xi}}{\ddot{\zeta}+1}\right) \tag{2.38}$$

then

$$\sin x_3 = \frac{\ddot{\xi}}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}} \quad \text{and} \quad \cos x_3 = \frac{\ddot{\zeta}+1}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}}. \tag{2.39}$$

Then, the expressions of x_1 and x_2 are deduced from (2.37)*:*

$$x_1 = \xi - \varepsilon\frac{\ddot{\xi}}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}} \quad \text{and} \quad x_2 = \zeta - \varepsilon\frac{\ddot{\zeta}+1}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}}. \tag{2.40}$$

Moreover, the inputs u_1 and u_2 are given by:

$$u_1 = \ddot{x}_1 = \frac{d^2}{dt^2}\left(\xi - \varepsilon\frac{\ddot{\xi}}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}}\right) \tag{2.41}$$

and

$$u_2 = \ddot{x}_2 = \frac{d^2}{dt^2}\left(\zeta - \varepsilon\frac{\ddot{\zeta}+1}{\sqrt{(\ddot{\xi})^2 + (\zeta^2+1)^2}}\right). \tag{2.42}$$

Thus, all state and input variables of the system are expressed as a function of ξ, ζ and their successive derivatives up to the order 4*, which proves that the system* (2.34) *is flat with (ξ, ζ) a flat output.*

Remark 2.4.1 *In Kaminski, Lévine, and Ollivier (2018), the authors have shown that the property of flatness is not defined globally. The Lie-Bäcklund isomorphisms φ and ψ are only locally defined, i.e. there might exist points in $\mathfrak{X}_0$ where no such isomorphisms exist or, otherwise stated, where the system is not flat. Such points are called intrinsic singularities. It has also been proven that the set of intrinsic singularities contains the set of equilibrium points of the system that are not first-order controllable.*

2.4.2 Linear flat system

As presented in Section 2.2, every linear controllable system of the form (2.5) can be transformed, by change of coordinates, into a canonical form (2.8). Therefore,

conditions of flatness and controllability for linear systems are equivalent according to the following theorem:

Theorem 2.4.1 (Fliess et al., 1993) *A linear system is flat if, and only if, it is controllable.*

In the sequel, we denote by $\mathbb{R}[\frac{d}{dt}]^{p\times q}$ the set of $p \times q$ matrices whose entries are polynomials in $\frac{d}{dt}$ with real coefficients:

$$a_0 + a_1\frac{d}{dt} + a_2\frac{d^2}{dt^2} + \ldots + a_k\frac{d^k}{dt^k}, \quad a_i \in \mathbb{R}. \tag{2.43}$$

The definition of the flatness in the context of linear systems is then given by:

Definition 2.4.2 (Lévine and Nguyen, 2003) *The linear system* (2.5) *is called differentially flat if, and only if, there exists a vector* $z = (z_1, \ldots, z_m) \in \mathbb{R}^m$ *such that:*

$$x_i = \sum_{j=1}^{m}\sum_{k=0}^{q_j} a_{i,j,k} z_j^{(k)} \qquad i = 1, \ldots, n \tag{2.44}$$

$$u_i = \sum_{j=1}^{m}\sum_{k=0}^{q_j+1} b_{i,j,k} z_j^{(k)} \qquad i = 1, \ldots, m, \tag{2.45}$$

z is also a linear combination of x, u and successive time derivatives of u.

In polynomial matrix language, equations (2.44) and (2.45) can be expressed as follows

$$x = Pz \quad \text{and} \quad u = Qz \tag{2.46}$$

where $P \in \mathbb{R}[\frac{d}{dt}]^{n\times m}$ and $Q \in \mathbb{R}[\frac{d}{dt}]^{m\times m}$. The vector z is called flat output and the matrices P and Q are called defining matrices.

2.5 Generation of a reference trajectory

Generating a reference trajectory is equivalent to predicting the motion of the system. It is based on the construction of a trajectory, in open-loop while respecting the constraints and for ideal conditions, so that the system reaches a sequence of points, and the calculation of the associated control inputs that generate this trajectory. The

differential flatness property ensures, in a simple way, the generation of a reference trajectory of a system.

Consider the following nonlinear system:

$$\dot{x} = f(x, u), \tag{2.47}$$

we want the system to move from the point

$$x(t_i) = x_i, \quad u(t_i) = u_i \tag{2.48}$$

at initial time t_i to the point

$$x(t_f) = x_f, \quad u(t_f) = u_f \tag{2.49}$$

at final time t_f. Trajectory constraints of type $(x(t), u(t)) \in A(t)$, where $A(t)$ is a sub-manifold of the manifold $X \times \mathbb{R}^m$ could be added to the motion planning initial problem (Levine, 2009). In general, the problem of finding a reference trajectory is quite challenging. There exist numerical methods that build an iterative solution: starting from the initial condition $t \mapsto (x_0(t), u_0(t))$, the system (2.47) is integrated and then the solution is verified against the final conditions. The mechanism is then repeated for $t \mapsto (x_1(t), u_1(t))$.

The flatness property ensures the calculation of the reference trajectory without having to approximate or solve any differential system. In fact, the flat output $z = (z_1, \ldots, z_m)$ does not satisfy any differential equation, thus it suffices to generate a reference trajectory of z. Then, reference trajectories of x and u can be deduced by flatness using equation (2.30).

Flatness-based trajectory planning

Suppose that the system (2.47) is flat with $z = (z_1, \ldots, z_m)^T$ a flat output. Then, according to Definition 2.4.1, the full state and input vectors can be written in function

of z and its successive derivatives:

$$x = \varphi_0(z, \dot{z}, \ldots, z^{(\rho)}) \quad \text{and} \quad u = \varphi_1(z, \dot{z}, \ldots, z^{(\rho+1)}). \tag{2.50}$$

Starting from initial and final conditions (2.48) and (2.49), we compute the initial and final conditions of the flat output z by the surjectivity of (φ_0, φ_1):

$$z(t_i) = z_i \quad \text{and} \quad z(t_f) = z_f. \tag{2.51}$$

Once a reference trajectory $t \mapsto z(t)$ for the flat output z is determined, we deduce the reference trajectories $t \mapsto (x(t), u(t))$ of x and u using (2.50). Recall that these trajectories identically satisfy the system of differential equations (2.47) and that the reference trajectory $z(t)$ of z must be at least $\rho + 1$ times differentiable.

The reference trajectory of the flat output $t \mapsto z(t)$ does not satisfy any differential equation. Thus, it can be constructed using *polynomial interpolation*. For more details on the generation of reference trajectories see Levine (2009, Chapter 7).

Example 2.5.1 *Back to the example of the non-holonomic vehicle, the flat output of the system is given by* $z = (x, y)^T = (z_1, z_2)^T$. *Suppose that we want the system to go from point* $(x_i, y_i) = (0, 0)$ *to point* $(x_f, y_f) = (3, 5)$ *in* $t_f = 300$s, *then we have:*

$$\begin{aligned} z_1(t_i) &= 0, \quad z_2(t_i) = 0, \\ z_1(t_f) &= 3, \quad z_2(t_f) = 5, \end{aligned} \tag{2.52}$$

which are the initial and final conditions of the flat output. We also add to this problem some constraints: we suppose that the starting and ending point are equilibrium points, i.e.

$$\begin{aligned} \dot{x}_i &= 0, \quad \dot{y}_i = 0, \\ \ddot{x}_i &= 0, \quad \ddot{y}_i = 0. \end{aligned}$$

In order to construct a reference trajectory of the flat output z, *we use a fifth-order polynomial interpolation:*

$$z(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5. \tag{2.53}$$

The reason for choosing order 5 *is because, according to the expression of the flatness* (2.20)*, we need to derive the flat output at least twice (see Levine (2009, Chapter 7)). It is now sufficient to replace the initial and final conditions in* (2.53)*, by also taking into account the constraints, to calculate the coefficients of* $z(t)$*. Figure 2.3 shows the reference trajectories of the flat output* $z = (x, y)$ *and of the state* θ*. The reference trajectory of the control inputs* (u, φ) *are illustrated in Figure 2.4. By applying the reference control inputs* $(u(t), \varphi(t))$*, computed by flatness, on the system* (2.17)*, the output of the system* (x, y, θ) *matches with the reference trajectories* $(x(t), y(t), \theta(t))$*, see Figure 2.5.*

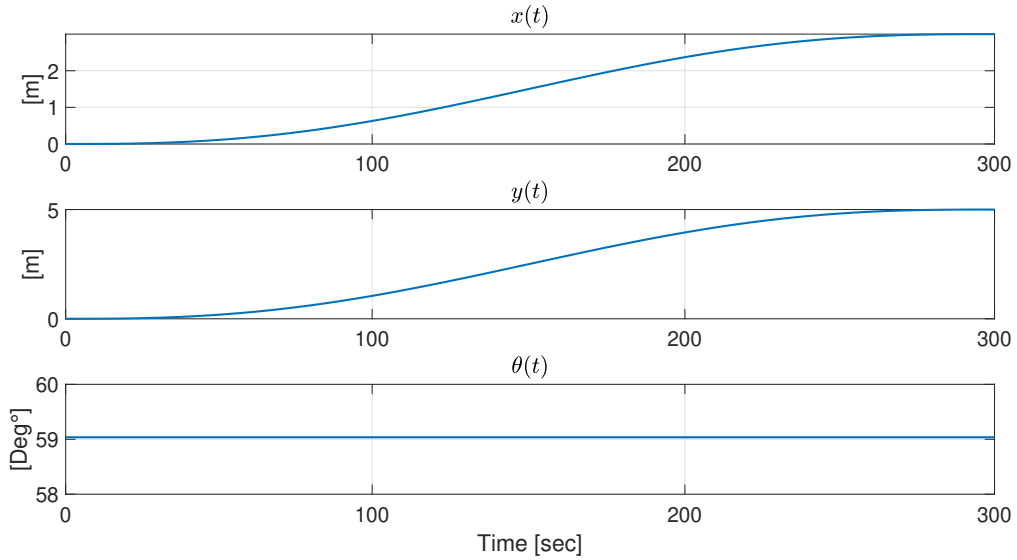

FIGURE 2.3: *Reference trajectory of the flat output* $z = (x, y)$ *and the state* θ

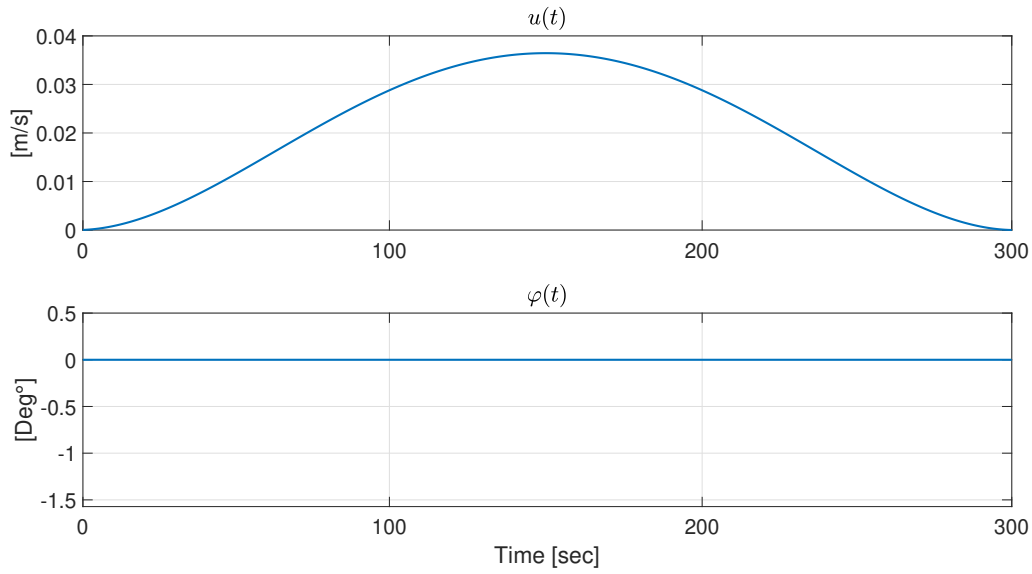

FIGURE 2.4: *Reference trajectory of the control input* (u, φ)

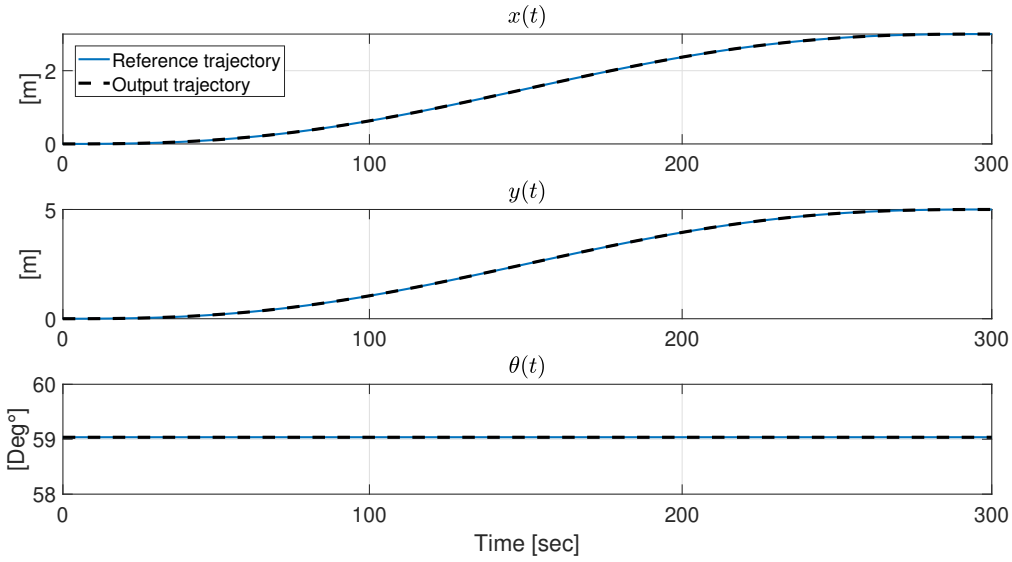

FIGURE 2.5: *Reference trajectories vs. outputs of the system*

2.6 Conclusion

This chapter presented a general overview on the appearance of the flatness property for the class of nonlinear dynamic systems. Starting from the property of controllability of linear systems, the differential flatness is one of the extensions of this property to nonlinear systems. Then, every linear controllable system is a flat system. Moreover, every nonlinear system that can be transformed, by a change of

coordinates, to a canonical form is a flat system. The change of coordinates is a Lie-Bäcklund isomorphism and the new state vector in the flat system is called flat output vector.

An important application of the flatness property of nonlinear systems is the generation of reference trajectories. That is, since the reference trajectory of the flat output does not satisfy any differential equation, it can be generated using polynomial interpolation. Consequently, reference trajectories of the system states, inputs and also outputs can be deduced from the reference trajectory of the flat output by simply differentiating the latter a finite number of time.

In addition to the simplicity that flatness provides for generating a reference trajectory, this property has many other applications in control theory such as *trajectory tracking* (see Antritter, Müller, and Deutscher (2004), Stumper, Svaricek, and Kennel (2009), Levine (2009)) and *designing robust controllers* (see Cazaurang (1997), Lavigne (2003)). The flatness property has also been proven to be efficient for the FTC (see Martínez-Torres et al. (2014), Torres (2014)) and FDI (see Mai, Join, and Reger (2006), Mai, Join, and Reger (2007), Suryawan, De Doná, and Seron (2010)).

The center of interest in the flatness is the calculation of flat outputs. For example, in the flatness-based FTC and FDI methods developed in Torres (2014), the flat outputs are used to generate redundant variables such as redundant states, redundant inputs and redundant outputs. Several methods of computation of flat outputs have been developed, we cite among them the *Smith Decomposition* and the *Unimodular Completion Algorithm*. In the next chapter, a recall on the computation methods of flat outputs is presented, as well as a contribution on the relation that exists between the different flat outputs of a flat system. Furthermore, an extension of the flatness-based FDI method, given by Torres (2014), is detailed with a characterization of the flat outputs that are useful for the FDI process.

Chapter 3

Characterization of Flat Outputs for Fault Detection and Isolation

Abstract

The flatness property of dynamic systems is based on the calculation of the flat outputs. There are several methods of computing of these flat outputs, among them there is the Smith decomposition method, also named diagonal decomposition method, of polynomial matrices and the unimodular completion algorithm. The applications of the flatness property are numerous, among them there is the application for the detection and isolation of faults on system sensors and actuators. In this chapter, a characterization of the flat outputs is presented for the purpose of nonlinear system diagnostics, especially the fault detection and isolation.

3.1 Introduction

The FDI problem has been introduced to automatic control as a paradigm for designing algorithms able to detect the outbreak of faults and isolate their causes. In Chapter 1, we have briefly recalled the FDI methods that exist in the literature. Among them, there are methods based on the flatness property of dynamic systems. The most recent of these flatness-based FDI methods is the one developed by Torres (2014). This method is applicable to linear and nonlinear systems, for both additive and multiplicative faults.

The residual signal, usually denoted by $r(t)$, is generated by the difference between the measured variable and its redundancy, calculated using the flat output and its successive time derivatives. In practice, the treatment of these residues is adapted, in the presence of noise, by introducing a threshold for each residue (Martínez-Torres et al., 2013a; Torres, 2014). The detectability of faults is simple: if at least one residue exceeds its threshold, then a fault is detected. Otherwise, no fault is detected. However, the isolability can take several forms according to the FDI method (Kóscielny et al., 2016).

Therefore, in this chapter, the theoretical isolability property of the method developed by Torres (2014) is analysed: every sensor and actuator admits a fault alarm signature, *i.e.* a number of residues affected by a fault. This fault alarm signature is defined beforehand based on the expression of the flat output. If this signature is distinct, then the associated fault is isolable. However, often, the full isolability of faults can not be achieved using only one flat output. So, we show that it is possible to increase the isolability of faults by considering multiple flat outputs, given that they are independent, hence introducing a characterization of the flat outputs for the FDI process, and rigorously completing some heuristic results from Martínez-Torres et al. (2013b).

This flatness-based FDI method is based on the expressions of the flat outputs for the purpose of generating redundant signals and, consequently, the residual vectors. In Section 3.2 of this chapter, we recall two methods of computation of flat outputs, and present a contribution to the relation that putatively exists between different flat outputs. Then, in Section 3.3, we present a generalization of

the flatness-based FDI method of Torres (2014), followed by an introduction to the so-called signature matrix and fault alarm signature for the aim of fault detection and fault isolation, in Section 3.4. As the full isolability of faults can sometimes not be ensured using only one flat output, in Section 3.5, we show that it is possible to increase the isolability of faults by considering multiple flat outputs, given the condition that they are independent, which leads to the characterization of the flat outputs for the aim of FDI.

3.2 Flat output computation

Recall that a system is said to be flat if, and only if, there exists a vector, called flat output, such that all the state, input and output variables can be expressed as functions of this flat output and its successive time derivatives. To our knowledge, there exist several methods of computation of the flat outputs. Among them there are the *Smith Decomposition* (Lévine, 2006) and the *Unimodular Completion Algorithm* (Fritzsche et al., 2016a). These two methods start from the implicit form (2.27) of the system and then move to the tangent system by applying differentiation to the implicit system. In this section, we present these two methods, and we show, using the Smith decomposition, that two different *tangent flat outputs*[1] can be related either differentially or algebraically. The unimodular completion algorithm itself allows, in some particular cases, a direct computation of flat outputs, *i.e.* without the need for complex calculations.

3.2.1 Smith decomposition

The Smith decomposition method for computing flat outputs is based on the diagonal decomposition of polynomial matrices. Moreover, it provides a necessary and sufficient condition for the flatness of nonlinear dynamic systems. This method has been developed for the classes of linear and nonlinear systems. Additionally, it has been extended to include the class of fractional linear systems as recalled later in Chapter 5.

[1] Flat outputs of the tangent system.

Consider the following implicit system

$$F(x, \dot{x}) = 0, \tag{3.1}$$

where $x = (x_1, \ldots, x_n) \in X$ is the state vector, F is a $\mathcal{C}^\infty$ mapping from TX to $\mathbb{R}^{n-m}$ and $rank\left(\frac{\partial F}{\partial \dot{x}}\right) = n - m$. A necessary and sufficient condition for the flatness of the system (3.1) is given by the following theorem:

Theorem 3.2.1 (Levine, 2009) *The implicit system* (3.1) *is locally flat at* $(\overline{x}_0, \overline{z}_0)$*, with* $\overline{x}_0 \in \mathfrak{X}_0$ *and* $\overline{z}_0 \in \mathbb{R}^m_\infty$ *if, and only if, there exists a Lie-Bäcklund isomorphism* $\varphi = (\varphi_0, \varphi_1, \ldots) = (\varphi_{0,1}, \ldots, \varphi_{0,n}, \varphi_{1,1}, \ldots, \varphi_{1,n}, \ldots)$ *from a neighborhood of* $\overline{z}_0$ *in* $\mathbb{R}^m_\infty$ *to a neighborhood of* $\overline{x}_0$ *in* $\mathfrak{X}_0$*, with inverse* $\psi = (\psi_0, \psi_1, \ldots)$*, satisfying* $\varphi(\overline{x}_0) = \overline{z}_0$ *and such that, locally:*

$$P(F)P(\varphi_0)dz = 0, \tag{3.2}$$

where

$$P(F) = \begin{pmatrix} \frac{\partial F_1}{\partial x_1} + \frac{\partial F_1}{\partial \dot{x}_1}\frac{d}{dt} & \cdots & \frac{\partial F_1}{\partial x_n} + \frac{\partial F_1}{\partial \dot{x}_n}\frac{d}{dt} \\ \vdots & & \vdots \\ \frac{\partial F_{n-m}}{\partial x_1} + \frac{\partial F_{n-m}}{\partial \dot{x}_1}\frac{d}{dt} & \cdots & \frac{\partial F_{n-m}}{\partial x_n} + \frac{\partial F_{n-m}}{\partial \dot{x}_n}\frac{d}{dt} \end{pmatrix}, \tag{3.3}$$

$$P(\varphi_0) = \begin{pmatrix} \sum_{j\geq 0} \frac{\partial \varphi_{0,1}}{\partial z_1^{(j)}}\frac{d^j}{dt^j} & \cdots & \sum_{j\geq 0} \frac{\partial \varphi_{0,1}}{\partial z_m^{(j)}}\frac{d^j}{dt^j} \\ \vdots & & \vdots \\ \sum_{j\geq 0} \frac{\partial \varphi_{0,n}}{\partial z_1^{(j)}}\frac{d^j}{dt^j} & \cdots & \sum_{j\geq 0} \frac{\partial \varphi_{0,n}}{\partial z_m^{(j)}}\frac{d^j}{dt^j} \end{pmatrix} \quad \text{and} \quad dz = \begin{pmatrix} dz_1 \\ \vdots \\ dz_m \end{pmatrix}. \tag{3.4}$$

Entries of matrices $P(F)$ *and* $P(\varphi_0)$ *are* $\frac{d}{dt}$*-polynomials whose coefficients are* $\mathcal{C}^\infty$ *functions, and the vector* dz *is called the vector of* 1*-forms*[2].

The proof of this theorem is detailed in Levine (2009, Chapter 6).

In order to compute the flat output vector z, we have to solve the variational system (3.2). In fact, from (3.2), it can be seen that the matrix $P(\varphi_0)$ sends the

[2]The differential or exterior derivative df of a function f is called 1-form. For a rigorous definition see Levine (2009, Chapter 2)

vector of 1-form dz to the kernel of $P(F)$, and has $P(\psi_0)$ as its inverse matrix, *i.e.* $P(\psi_0)P(\varphi_0) = I_m$. Then, the vector of 1-form dz can be expressed as follows:

$$dz = P(\psi_0)dx. \tag{3.5}$$

Since the vector z is a flat output, the components of the vector dz are linearly independent, and we have:

$$P(F)P(\varphi_0)dz = 0 \quad \Rightarrow \quad P(F)P(\varphi_0) = 0. \tag{3.6}$$

Then, the matrix $P(\varphi_0)$ is a solution of the equation

$$P(F)\Theta = 0 \tag{3.7}$$

and the matrix $P(\psi_0)$ is a solution of

$$Q\Theta = I_m. \tag{3.8}$$

Notice that $\Theta \in \ker(P(F))$. This kind of solution is valid for matrices on a ring. However, the entries of $P(F)$ are polynomials in $\frac{d}{dt}$, whose coefficients are $\mathcal{C}^\infty$ functions, that do not form a domain. Then, we restrict to the ring of polynomials whose coefficients are meromorphic functions [3] on $\mathfrak{X}_0$, denoted by $\mathcal{R}[\frac{d}{dt}]$, and we suppose that F is a meromorphic function on $\mathfrak{X}_0$. In this case, the matrix $P(F) \in \mathcal{M}_{(n-m)\times n}[\frac{d}{dt}]$, the ring of polynomial matrices with entries in $\mathcal{R}[\frac{d}{dt}]$. Moreover, an invertible matrix U in $\mathcal{M}_{p\times p}[\frac{d}{dt}]$ whose inverse is also in $\mathcal{M}_{p\times p}[\frac{d}{dt}]$ is called unimodular matrix. The set of unimodular matrices is denoted by $\mathcal{U}_p[\frac{d}{dt}]$.

Matrices in $\mathcal{M}_{p\times q}[\frac{d}{dt}]$ have a specific property: they can be diagonalized according to the Smith decomposition as follows (Cohn, 1971):

Theorem 3.2.2 (Smith decomposition) *Let $M \in \mathcal{M}_{p\times q}[\frac{d}{dt}]$, then there exist unimodular matrices $U \in \mathcal{U}_q[\frac{d}{dt}]$ and $V \in \mathcal{U}_p[\frac{d}{dt}]$ and a diagonal matrix Δ of size $r \times r$ with $r = \min(p, q)$ whose diagonal elements $\delta_{i,i} \in \mathcal{R}[\frac{d}{dt}]$, and verifying $\delta_{i,i}$ divides $\delta_{j,j}$, for all*

[3] A meromorphic function is a rational fraction of analytical functions.

$1 < i < j \leq r$, *such that:*

$$VMU = \begin{cases} (\Delta, 0_{p,q-p}) & \text{if} \quad p < q \\ \begin{pmatrix} \Delta \\ 0_{p-q,q} \end{pmatrix} & \text{if} \quad p > q \end{cases} . \tag{3.9}$$

We say that $V \in$ *L-Smith(M) and* $U \in$ *R-Smith(M).*

These matrices can also have the property of hyper-regularity given by the following definition:

Definition 3.2.1 (Hyper-regularity) *A matrix* $M \in \mathcal{M}_{p\times q}[\frac{d}{dt}]$ *is hyper-regular if, and only if, there exists two unimodular matrices* $U \in \mathcal{U}_q[\frac{d}{dt}]$ *and* $V \in \mathcal{U}_p[\frac{d}{dt}]$ *such that:*

$$VMU = \begin{cases} (I_r, 0_{p,q-p}) & \text{if} \quad p < q \\ \begin{pmatrix} I_r \\ 0_{p-q,q} \end{pmatrix} & \text{if} \quad p > q \end{cases} \tag{3.10}$$

with $r = \min(p, q)$.

The main property of the differential flatness is illustrated in the following proposition:

Proposition 3.2.1 (Lévine, 2011) *If the system* (3.1) *is flat at a point* $\overline{x}_0 \in \mathfrak{X}_0$*, then there exists a neighbourhood* $\mathcal{V}$ *of* $\overline{x}_0$ *where* $P(F)$ *is hyper-regular.*

3.2.1.1 Computation procedure

From now on, we suppose that the matrix $P(F) \in \mathcal{M}_{(n-m)\times n}[\frac{d}{dt}]$ is hyper-regular in a neighborhood $\mathcal{V}$ of $\overline{x}_0$, then, according to Definition 3.2.1, there exists a unimodular matrix $U \in \mathcal{U}_n\left[\frac{d}{dt}\right]$ such that:

$$P(F)U = \begin{pmatrix} I_{(n-m)} & 0_{(n-m)\times m} \end{pmatrix}. \tag{3.11}$$

Let $\widehat{U} \triangleq U \begin{pmatrix} 0_{(n-m)\times m} \\ I_m \end{pmatrix}$ be the last m columns of U, then $P(F)\widehat{U} = 0$ and $\widehat{U}$ is a solution of (3.7). Moreover, let $\widehat{Q} \in$ L-Smith($\widehat{U}$) of size $n \times n$:

$$\widehat{Q}\,\widehat{U} = \begin{pmatrix} I_m \\ 0_{(n-m)\times m} \end{pmatrix}, \tag{3.12}$$

and let $Q \triangleq (I_m \;\; 0_{m\times(n-m)})\,\widehat{Q}$ be the first m rows of $\widehat{Q}$, then

$$Q\widehat{U} = (I_m \;\; 0_{m\times(n-m)}) \begin{pmatrix} I_m \\ 0_{(n-m)\times m} \end{pmatrix} = I_m. \tag{3.13}$$

Hence,

$$Q = \widehat{U}^{\dagger} = (0_{m\times(n-m)} \;\; I_m)U^{-1} \tag{3.14}$$

is the pseudo-inverse of $\widehat{U}$ and a solution of (3.8). Further details can be found in Levine (2009, Chapter 6).

Let $\omega = (\omega_1, \ldots, \omega_m)^T$ be the vector of 1-forms defined by

$$\omega \triangleq \widehat{U}^{\dagger} dx. \tag{3.15}$$

Then, the necessary and sufficient condition (3.2) of the flatness becomes:

Theorem 3.2.3 (Levine, 2009) *The system* (3.1) *is flat if, and only if, the vector of 1-forms* ω*, given by* (3.15)*, is integrable, i.e. there exists a unimodular matrix* $M \in \mathcal{U}_m[\frac{d}{dt}]$ *such that*

$$d(M\omega) = 0. \tag{3.16}$$

In this case, the flat output z *is given by:*

$$dz = M\omega. \tag{3.17}$$

The vector ω is called *tangent flat output* of the variational system (3.2). Conditions of integrability of ω are detailed in Levine (2009, Chapter 6).

3.2.1.2 Smith decomposition: contributions and observations

The Smith decomposition of the matrix $P(F)$ is not unique (Levine, 2009). In fact, if $U \in \mathcal{U}_n[\frac{d}{dt}]$ is in R-Smith$(P(F))$ then, the matrix MU with $M \in \mathcal{U}_n[\frac{d}{dt}]$ is also in R-Smith$(P(F))$. Therefore, a flat system admits an infinite number of flat outputs. According to this, a property on the relation that exists between different tangent flat outputs is given by the following proposition:

Proposition 3.2.2 *Let $\Omega_{\overline{x}}$ be the set of all tangent flat outputs at $\overline{x}$ of a flat system. Then, for all ω_1 and $\omega_2 \in \Omega_{\overline{x}}$, there exists a unimodular matrix $\mathcal{K} \in \mathcal{U}_m[\frac{d}{dt}]$ such that*

$$\omega_1 = \mathcal{K}\ \omega_2. \tag{3.18}$$

Proof. The matrix $P(F)$ is hyper-regular, then there exists a hyper-regular matrix $\widehat{U}$ such that

$$\omega = \widehat{U}^{\dagger} dx \tag{3.19}$$

and then $dx = \widehat{U}\omega$. Let $\omega_1 = \widehat{U}_1^{\dagger} dx$ and $\omega_2 = \widehat{U}_2^{\dagger} dx$ be two different tangent flat outputs at $\overline{x}$, then

$$\omega_1 = \widehat{U}_1^{\dagger}\widehat{U}_2\ \omega_2 \triangleq \mathcal{K}\ \omega_2 \tag{3.20}$$

with

$$\mathcal{K} = \widehat{U}_1^{\dagger}\widehat{U}_2. \tag{3.21}$$

Let us prove that $\mathcal{K}$ is unimodular. $\widehat{U}_1^{\dagger}$ and $\widehat{U}_2$ are hyper-regular by construction, then if ξ is a vector such that $\widehat{U}_1^{\dagger}\widehat{U}_2\xi = 0$ with $\xi \neq 0$ then $\zeta = \widehat{U}_2\xi$ is also $\neq 0$ and $\widehat{U}_1^{\dagger}\zeta = 0$ which contradicts the hyper-regularity of $\widehat{U}_1^{\dagger}$. Hence, $\mathcal{K}$ is hyper-regular and square, then unimodular. □

3.2.1.3 Computational tools

In Verhoeven (2016), a Maple-toolbox named **DifferentialForms** has been developed in order to determine whether a nonlinear system is flat or not. It also offers the capability to compute multiple flat outputs using the algorithm of Smith decomposition, based on the minimal-basis decomposition (Antritter and Middeke, 2011).

In spite of the important development of this toolbox, according to our observations and experiences, it is still not possible to compute flat outputs for an arbitrary flat system, because of the symbolic computations of the Smith decomposition of matrices, which takes a lot of memory space and computation time, and the resolution of partial differential equations for the problem of integrability, which returns arbitrary functions (see pdsolve-Maple).

For the non-holonomic vehicle example 2.4.1, the toolbox works very well in the step of Smith decomposition, but for the computation of the matrix M of integrability (see equation (3.16)), the result that we obtain is a matrix whose entries are arbitrary functions of the state vector (x, y, θ). The toolbox has been also tested on the example of the inverse pendulum 2.4.2, where $P(F)$ has degree 2 (see Levine, 2009). We noticed that the toolbox is incapable of calculating the Smith decomposition of $P(F)$. Thus, the toolbox still requires more development and simplifications so that it becomes more user-friendly.

3.2.1.4 Smith decomposition for linear flat systems

The same procedure of computations can be applied for the class of linear systems:

$$\dot{x} = Ax + Bu. \tag{3.22}$$

First, we transform the explicit system (3.22) into an implicit system by left multiplying it by a $n \times (n - m)$ matrix C of full column rank, such that $C^T B = 0$:

$$C^T(\dot{x} - Ax) = 0 \tag{3.23}$$

which is equivalent to

$$C^T(I_n\frac{d}{dt} - A)x = 0 \tag{3.24}$$

Thus, $P(F) \triangleq C^T(I_n\frac{d}{dt} - A)$. Then, supposing that the system (3.22) is controllable, *i.e.* flat, a solution of the system $P(F)\Theta = 0$ is given by:

$$\Theta = U \begin{pmatrix} 0_{n-m,m} \\ \\ I_m \end{pmatrix} = \widehat{U} \tag{3.25}$$

with $U \in$ R-Smith$(P(F))$. Let $\widehat{Q} \in$ L-Smith$(\widehat{U})$, and $Q = (I_m \;\; 0_{m,n-m})\widehat{Q}$, then the flat output z is given by:

$$z = Qx \quad \text{and conversely} \quad x = \widehat{U}z. \tag{3.26}$$

The input u can also be deduced directly as function of $(z, \dot{z}, \ldots)$ using the expression of the linear system (3.22). This computation algorithm of flat outputs for linear systems can also be applied directly to the explicit system (3.22). For more details, see Lévine and Nguyen (2003).

3.2.2 Computation via unimodular completion

Another algorithm of computation of flat outputs has been developed in Fritzsche et al. (2016a) for the classes of linear and nonlinear flat systems. This algorithm is based on the notion of unimodular completion of $\frac{d}{dt}$-polynomial matrices. It differs from the Smith decomposition in that this algorithm extracts first coefficient matrices from $\frac{d}{dt}$-polynomial matrices, and then computes with the coefficients matrix to get flat outputs.

3.2.2.1 Preliminary definitions

This algorithm is based on the notion of unimodular completion of matrices in $\mathcal{M}_{p\times q}[\frac{d}{dt}]$:

Definition 3.2.2 *Given a hyper-regular matrix $M \in \mathcal{M}_{p\times q}\left[\frac{d}{dt}\right]$ with $p \leq q$, we say that $N \in \mathcal{M}_{(q-p)\times q}\left[\frac{d}{dt}\right]$ is a unimodular completion of M if, and only if,*

$$\begin{pmatrix} M \\ N \end{pmatrix} \in \mathcal{U}_q\left[\frac{d}{dt}\right]. \tag{3.27}$$

Proposition 3.2.3 (Fritzsche et al., 2016b) *The vector ω such that $\omega = Qdx$, with $Q \in \mathcal{M}_{m\times n}[\frac{d}{dt}]$, is a tangent flat output of the variational system (3.2) if, and only if, Q is a unimodular completion of $P(F)$.*

Let $\widehat{U}^\dagger$, defined by (3.14), be the matrix calculated by the Smith decomposition. This matrix admits the following property:

Proposition 3.2.4 *The matrix $\widehat{U}^\dagger$ is a unimodular completion of $P(F)$.*

Proof. The matrix $P(F)$ is hyper-regular, then there exists $U \in \mathcal{U}_n[\frac{d}{dt}]$ such that

$$P(F)U = \begin{pmatrix} I_{(n-m)} & 0_{(n-m)\times m} \end{pmatrix}, \tag{3.28}$$

then $P(F) = \begin{pmatrix} I_{n-m} & 0_{(n-m)\times m} \end{pmatrix} U^{-1}$ constitutes the first $n-m$ rows of the matrix U^{-1}, *i.e.*

$$U^{-1} = \begin{pmatrix} P(F) \\ W \end{pmatrix} \tag{3.29}$$

with $W \in \mathcal{M}_{m\times n}[\frac{d}{dt}]$. Moreover, according to (3.14), the matrix $\widehat{U}^\dagger$ constitutes the last m rows of U^{-1}, hence

$$U^{-1} = \begin{pmatrix} P(F) \\ \widehat{U}^\dagger \end{pmatrix}, \tag{3.30}$$

which proves that $\widehat{U}^\dagger$ is a unimodular completion of $P(F)$. □

Then, instead of calculating $\widehat{U}^\dagger$ by the Smith decomposition, we can calculate it by

the unimodular completion algorithm, presented below. Recall that the tangent flat output is given by $\omega = \widehat{U}^{\dagger} dx$ and the flat output of the system is given by $dz = M\omega$, as long as ω is integrable.

3.2.2.2 Computation procedure

The unimodular completion algorithm is iterative and consists of three steps: *Reduction*, *Zero-space Decomposition* and *Elimination*. The starting point is the variational system (3.2) given by:

$$P(F)dx = 0 \tag{3.31}$$

with $dx = P(\varphi_0)dz$. The matrix $P(F) \in \mathcal{M}_{(n-m)\times n}[\frac{d}{dt}]$ is divided into two coefficient matrices $P_{0,[0]} \triangleq \frac{\partial F}{\partial x}(x, \dot{x})$ and $P_{1,[0]} \triangleq \frac{\partial F}{\partial \dot{x}}(x, \dot{x})$. Then, the system (3.31) becomes:

$$0 = \Big(P_{0,[0]} + P_{1,[0]}\frac{d}{dt}\Big)v_{[0]} \tag{3.32}$$

with $v_{[0]} = dx$. The index in the bracket indicates the iteration number.

Remark 3.2.1 *Since the derivative of the product of two functions f_1 and f_2 is given by*

$$\frac{d}{dt}f_1 f_2 = \dot{f}_1 f_2 + f_1 \dot{f}_2 = (\dot{f}_1 + f_1\frac{d}{dt})f_2 \tag{3.33}$$

then, a rule for the right shift of the operator via a time function is specified:

$$\frac{d}{dt}f_1 = \dot{f}_1 + f_1\frac{d}{dt}. \tag{3.34}$$

According to (3.34), equation (3.32) becomes:

$$0 = \Big(P_{0,[0]} + \frac{d}{dt}P_{1,[0]} - \dot{P}_{1,[0]}\Big)v_{[0]}. \tag{3.35}$$

Reduction:

Starting from

$$\Big(P_{0,[i]} + \frac{d}{dt}P_{1,[i]} - \dot{P}_{1,[i]}\Big)v_{[i]} = 0, \tag{3.36}$$

we consider the change of coordinates

$$v_{[i]} = P_{1,[i]}^{\dagger R} v_{[i+1]} + P_{1,[i]}^{\perp R} w_{[i+1]} \tag{3.37}$$

with $P_{1,[i]}^{\dagger R}$ is the right pseudo-inverse of $P_{1,[i]}$:

$$P_{1,[i]} P_{1,[i]}^{\dagger R} = I_{n-m} \tag{3.38}$$

and $P_{1,[i]}^{\perp R}$ is the right orthonormal:

$$P_{1,[i]} P_{1,[i]}^{\perp R} = 0. \tag{3.39}$$

By injecting (3.37) in (3.36), the system (3.36) in the new coordinates becomes:

$$0 = \dot{v}_{[i+1]} + A_{[i]} v_{[i+1]} + B_{[i]} w_{[i+1]} \tag{3.40}$$

with

$$A_{[i]} = (P_{0,[i]} - \dot{P}_{1,[i]}) P_{1,[i]}^{\dagger R} \quad \text{and} \quad B_{[i]} = (P_{0,[i]} - \dot{P}_{1,[i]}) P_{1,[i]}^{\perp R}. \tag{3.41}$$

The matrix $B_{[i]}$, being in $\mathcal{R}_{(n_i - m_i) \times m_i}$, the ring of meromorphic functions, two cases can be distinguished: if $rank\ (B_{[i]}) = r_i < m_i$ then a zero-space decomposition is needed to reduce the dimension. If $rank\ (B_{[i]}) = m_i$ *i.e.* $B_{[i]}$ is of full column rank we move on to the Elimination step.

Zero-space decomposition:

If $rank\ (B_{[i]}) = r_i < m_i$ we need to decompose the matrix $P_{1,[i]}^{\perp R}$ into the form

$$P_{1,[i]}^{\perp R} = (\widetilde{P}_{1,[i]}^{\perp R} \;\; Z_{[i]}) \tag{3.42}$$

such that the matrix $B_{[i]}$ becomes

$$B_{[i]} = (P_{0,[i]} - \dot{P}_{1,[i]})(\widetilde{P}_{1,[i]}^{\perp R} \;\; Z_{[i]}) = (\widetilde{B}_{[i]} \;\; 0) \tag{3.43}$$

with *rank* $(\widetilde{B}_{[i]}) = r_i$. For this purpose, the following matrices are introduced:

$$Z_{[i]} \triangleq P_{1,[i]}^{\perp R} B_{[i]}^{\perp R} \quad \text{and} \quad \widetilde{P}_{1,[i]}^{\perp R} \triangleq P_{1,[i]}^{\perp R} \Big(\big(B_{[i]}^{\perp R} \big)^{\perp L} \Big)^T \tag{3.44}$$

and, in this case, the change of coordinates (3.37) is replaced by:

$$v_{[i]} = P_{1,[i]}^{\dagger R} v_{[i+1]} + \widetilde{P}_{1,[i]}^{\perp R} w_{[i+1]} + Z_{[i]} z_{[i+1]}. \tag{3.45}$$

Then, the system (3.40) becomes:

$$0 = \dot{v}_{[i+1]} + A_{[i]} v_{[i+1]} + \widetilde{B}_{[i]} w_{[i+1]}. \tag{3.46}$$

Elimination:

Back to the Reduction step, if the matrix $B_{[i]}$ is not of full row rank, *i.e. rank* $B_{[i]} < n_i - m_i$, then the dimension of the system (3.36) must be reduced. For this purpose, we eliminate the variable $w_{[i+1]}$ from (3.40) by multiplying it by $B_{[i]}^{\dagger L}$, which leads to:

$$0 = \Big(P_{0,[i+1]} + P_{1,[i+1]} \frac{d}{dt} \Big) v_{[i+1]} \tag{3.47}$$

with

$$P_{0,[i+1]} = B_{[i]}^{\dagger L} A_{[i]} \quad \text{and} \quad P_{1,[i+1]} = B_{[i]}^{\dagger L}. \tag{3.48}$$

The dimension of the latter system is a reduced dimension of the system (3.36) and then the same procedure is repeated to the system (3.47). The calculations stop at iteration k when a full row rank of $B_{[k]}$ is reached.

Remark 3.2.2 *In the case where the zero-space decomposition is considered, the process of the Elimination step is applied to the equation* (3.46) *by replacing* $B_{[i]}^{\dagger L}$ *by* $\widetilde{B}_{[i]}^{\dagger L}$.

Construction of the unimodular completion:

In each iteration i, a relation between $v_{[i]}$ and $v_{[i+1]}$ can be deduced from (3.37), by multiplying it by $P_{1,[i]}$:

$$v_{[i+1]} = P_{1,[i]} v_{[i]} \tag{3.49}$$

and conversely from (3.40):

$$v_{[i]} = \left(P_{1,[i]}^{\dagger R} - P_{1,[i]}^{\perp R}\left(B_{[i]}^{\dagger L}\frac{d}{dt} - B_{[i]}^{\dagger L} A_{[i]}\right)\right) v_{[i+1]} \triangleq G_i\left(\frac{d}{dt}\right) v_{[i+1]}. \tag{3.50}$$

After a finite number $k+1$ of iterations, a relation between $v_{[k+1]}$ and $v_{[0]}$ is determined as follows:

$$v_{[k+1]} = P_{1,[k]} P_{1,[k-1]} \dots P_{1,[0]} v_{[0]} = Q v_{[0]} \tag{3.51}$$

and Q is the unimodular completion of the matrix $P(F)$. Moreover, we have

$$v_{[0]} = G_0\left(\frac{d}{dt}\right) G_1\left(\frac{d}{dt}\right) \dots G_k\left(\frac{d}{dt}\right) v_{[k+1]}. \tag{3.52}$$

Remark 3.2.3 *If the case where the zero-space decomposition is considered, the inverse of the equation* (3.45) *is given by:*

$$z_{[i+1]} = Z_{[i]}^{\dagger L} P_{1,[i-1]} \dots P_{1,[0]} v_{[0]} = \widehat{Q} v_{[0]} \tag{3.53}$$

where the matrix $Z_{[i]}^{\dagger L}$ *verifies:*

$$Z_{[i]}^{\dagger L} Z_{[i]} = I, \quad Z_{[i]}^{\dagger L} P_{1,[i]}^{\dagger R} = 0, \quad \text{and} \quad Z_{[i]}^{\dagger L} \widetilde{P}_{1,[i]}^{\dagger R} = 0. \tag{3.54}$$

Finally, Matrices Q *and* $\widehat{Q}$ *in equations* (3.51) *and* (3.53) *constitute the unimodular completion of the matrix* $P(F)$.

This algorithm has been implemented using Python. The readers can find the toolbox in Fritzsche (2016).

3.2.2.3 Direct flat representation

The unimodular completion method provides, in some particular cases, a direct representation of flat outputs:

Definition 3.2.3 (Pomet, 1997) *Let* (2.24) *be a flat system, it is called* $(-1)-$*flat or* $x-$*flat if, and only if, there is a flat output* z *such that* z *depends only on* x*, i.e.*

$$z = \psi(x). \tag{3.55}$$

Consider a subclass of $(-1)-$flat systems called direct flat systems defined as follows:

Definition 3.2.4 (Fritzsche et al., 2016b) *We say that a* $(-1)-$*flat system is a direct flat system if there exists a permutation* $\sigma : \{1, \dots, n\} \longmapsto \{\sigma(1), \dots, \sigma(n)\}$ *such that there exists a flat output given by* $z = (x_{\sigma(1)}, \dots, x_{\sigma(m)})$*. Such flat output is called direct flat output.*

Proposition 3.2.5 (Fritzsche et al., 2016b) *Let* $P(F)$ *defined by* (3.3) *be hyper-regular. Assume that there exists* Π*, a column permutation matrix such that*

$$\widetilde{P}(F) \triangleq P(F)\Pi = \begin{pmatrix} A & B \end{pmatrix} \tag{3.56}$$

with $A \in \mathcal{U}_{(n-m)}\left[\frac{d}{dt}\right]$ *and* $B \in \mathcal{M}_{(n-m)\times m}\left[\frac{d}{dt}\right]$*. Then denoting by* $\widetilde{H} = \begin{pmatrix} 0_{m\times(n-m)} & I_m \end{pmatrix}$ *and* $H = \widetilde{H}\Pi^T$*, which are constant matrices,* $\widetilde{H}$ *(resp.* H*) is a unimodular completion of* $\widetilde{P}(F)$ *(resp.* $P(F)$*). A tangent flat output* ω *is given by* $\omega = Hdx$ *and always satisfies the integrability condition,* i.e. $d\omega = 0$*. Hence, a (direct) flat output* z *of the non linear system is given by*

$$z = Hx. \tag{3.57}$$

The matrix $\widetilde{P}(F)$, defined in proposition 3.2.5, is called a direct flat representation, for which $\widetilde{z} \triangleq \widetilde{H}x$ is a direct flat output.

Example 3.2.1 *Consider the example of non-holonomic vehicle given in 2.3. After elimination of the two first equations from* (2.17)*, we obtain the following implicit system:*

$$F(x, \dot{x}) = \dot{x}\sin\theta - \dot{y}\cos\theta. \tag{3.58}$$

Its variation leads to:

$$0 = P(F)dX = \begin{pmatrix} \sin\theta\frac{d}{dt} & -\cos\theta\frac{d}{dt} & \dot{x}\cos\theta + \dot{y}\sin\theta \end{pmatrix} \begin{pmatrix} dx \\ dy \\ d\theta \end{pmatrix}. \tag{3.59}$$

Let $\Pi = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$ *be a permutation matrix, we obtain*

$$\widetilde{P}(F) = P(F)\Pi = \begin{pmatrix} \dot{x}\sin\theta + \dot{y}\cos\theta & -\cos\theta\frac{d}{dt} & \sin\theta\frac{d}{dt}. \end{pmatrix} \tag{3.60}$$

The matrix $\widetilde{P}(F)$ *is a direct flat representation, then, according to proposition 3.2.5, a unimodular completion of* $P(F)$ *is given by:*

$$H = \widetilde{H}\Pi^T = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \tag{3.61}$$

hence, a direct flat output is given by $z = Hx = (x, y)^T$.

To conclude this section, this method of computation of flat outputs has also been developed for the class of linear controllable systems. The reader is referred to the work of Fritzsche et al. (2016a). Moreover, this algorithm has been implemented

using Python (see UC algorithm). In Chapter 5 of this thesis, we prove that this method can be extended for the class of fractional linear flat systems by employing some properties of fractional calculus.

3.3 Flatness-based residual generation

The FDI technique is based on the generation of residual vectors, which represent the difference between the measured variables and their redundancies. In Martínez-Torres et al. (2014), the redundant variables are calculated using the flat outputs and their successive time derivatives.

Consider the following nonlinear system

$$\dot{x} = f(x, u) \tag{3.62}$$

where x, the vector of states, evolves in a n-dimensional manifold X, $u \in \mathbb{R}^m$ is the vector of inputs, $m \leq n$ and $rank\ (\frac{\partial f}{\partial u}) = m$. We suppose that the system (3.62) is flat with $z = (z_1, \ldots, z_m)$ as flat output.

On the contrary to the assumption stated by Martínez-Torres et al. (2014) that the full state is measured by sensors, *i.e.* the output vector y is equal to the state vector x, we suppose that the output vector y is an arbitrary function of x and u, of arbitrary dimension $p \geq m$:

$$y \triangleq (y_1, \ldots, y_p) = h(x, u). \tag{3.63}$$

Then, the components $y_1, \ldots, y_p$ of y are measured by sensors $\mathsf{S}_1, \ldots, \mathsf{S}_p$, respectively. We denote their measurements by:

$$y^s \triangleq (y_1^s, \ldots, y_p^s). \tag{3.64}$$

We also suppose that the flat output z is part of these measurements, and we denote it by z^s:

$$z^s = (z_1^s, \ldots, z_m^s) \triangleq \mathrm{pr}_{\mathbb{R}^m}(y^s), \tag{3.65}$$

and without loss of generality, we consider that the components of z^s are the first m components of y^s:

$$z^s = (y_1^s, \ldots, y_m^s)^T. \tag{3.66}$$

Moreover, the value of the input vector $u = (u_1, \ldots, u_m)^T$, corresponding to the actuators $\mathsf{A}_1, \ldots, \mathsf{A}_m$, is assumed to be available at every time.

According to the differential flat equation (2.30), the state and input read:

$$x^z = \varphi_0(\overline{z^s}^{(\rho)}) \quad \text{and} \quad u^z = \varphi_1(\overline{z^s}^{(\rho+1)}) \tag{3.67}$$

where the superscript z indicates that they are evaluated as functions of the measured flat output z^s. Vectors x^z and u^z are also called the redundant vectors associated to x and u, respectively. In addition, according to (3.63), the redundant output y_k^z, for $k = 1, \ldots, p$, computed via the measured flat output z^s and associated to the measured output y_k^s, is given by:

$$y_k^z \triangleq h_k(\varphi_0(\overline{z^s}^{(\rho)}), \varphi_1(\overline{z^s}^{(\rho+1)})). \tag{3.68}$$

Once the redundant variables are calculated, we can generate the residual signals according to the following definition:

Definition 3.3.1 (Residual generation) *The k^{th}-sensor residue R_{S_k} and l^{th}-input residue R_{A_l}, for $k = 1, \ldots, p$ and $l = 1, \ldots, m$, are given by:*

$$R_{\mathsf{S}_k} = y_k^s - y_k^z, \quad \textit{and} \quad R_{\mathsf{A}_l} = u_l - u_l^z. \tag{3.69}$$

respectively.

Figure 3.1 illustrates the flatness-based residual generation of our approach.

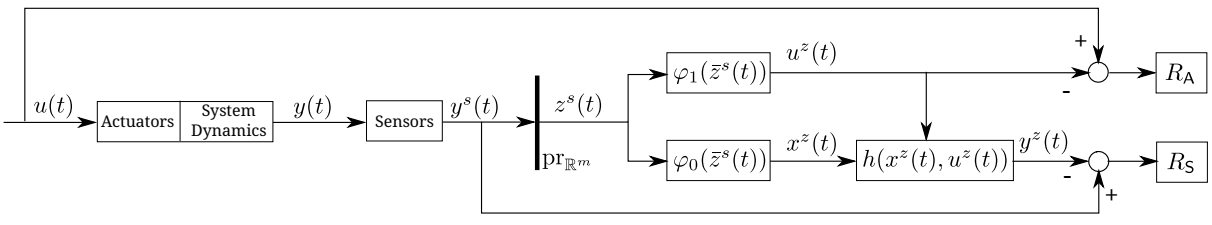

FIGURE 3.1: *Flatness-based residual generation*

In total, there exist $p + m$ residues for a single flat output z^s. The full residue vector is denoted by:

$$\begin{aligned} r &= (R_{\mathsf{S}_1}, \ldots, R_{\mathsf{S}_m}, R_{\mathsf{S}_{m+1}}, \ldots, R_{\mathsf{S}_p}, R_{\mathsf{A}_1}, \ldots, R_{\mathsf{A}_m})^T \\ &= (r_1, \ldots, r_m, r_{m+1}, \ldots, r_p, r_{p+1}, \ldots, r_{p+m})^T. \end{aligned} \tag{3.70}$$

Note that, according to the assumption (3.66), the first m components of y^z are equal to the corresponding components of z^s:

$$y^z = (z_1^s, \ldots, z_m^s, h_{m+1}(\varphi_0(\overline{z^s}), \varphi_1(\overline{z^s})), \ldots, h_p(\varphi_0(\overline{z^s}), \varphi_1(\overline{z^s})))^T. \tag{3.71}$$

Then, the first m residues $r_1, \ldots, r_m$, associated to sensors $\mathsf{S}_1, \ldots, \mathsf{S}_m$, are identically zero:

$$r_k = R_{\mathsf{S}_k} = y_k^s - y_k^z = z_k^s - z_k^s = 0, \quad \forall k = 1, \ldots, m. \tag{3.72}$$

Therefore, the residue vector (3.70) becomes:

$$\begin{aligned} r &= (0, \ldots, 0, R_{\mathsf{S}_{m+1}}, \ldots, R_{\mathsf{S}_p}, R_{\mathsf{A}_1}, \ldots, R_{\mathsf{A}_m})^T \\ &= (0, \ldots, 0, r_{m+1}, \ldots, r_p, r_{p+1}, \ldots, r_{p+m})^T. \end{aligned} \tag{3.73}$$

In fact, a residue which is always equal to zero indicates that it cannot be affected by faults on one of the sensors or actuators. Then, it is not useful for fault detection, and hence can be eliminated from the vector of residues (3.73), which is truncated to keep the last p components only. The truncated vector is denoted by r_τ:

$$\begin{aligned} r_\tau &= (R_{\mathsf{S}_{m+1}}, \ldots, R_{\mathsf{S}_p}, R_{\mathsf{A}_1}, \ldots, R_{\mathsf{A}_m})^T \\ &= (r_{\tau_1}, r_{\tau_2}, \ldots, r_{\tau_p})^T. \end{aligned} \tag{3.74}$$

Hypothesis: There is only one fault at a time affecting sensors or actuators.

Derivative estimation

The components $(z_1^s, \ldots, z_m^s)$ of the flat output z^s, must be differentiated in order to calculate the redundant values y^z and u^z. However, these derivatives may not exist because of the presence of noise on sensors and actuators. So, these derivatives have to be estimated.

Many methods have been developed in the literature. We cite among them the algebraic derivative estimation (Mboup, Join, and Fliess, 2007; Zehetner, Reger, and Horn, 2007) and high-gain observers (Vasiljevic and Khalil, 2008). In Martínez-Torres et al. (2014), for example, the derivatives are computed via a high-gain observer, possibly completed by a low-pass filter to improve its robustness.

3.4 Fault detection and isolation

In order to detect and isolate faults in this approach, the notion of *signature matrix* is introduced:

Definition 3.4.1 (Signature matrix) *Given the vector of residues r_τ defined in* (3.74) *and $\zeta = (y_1^s, \ldots, y_p^s, u_1, \ldots, u_m)^T \in \mathbb{R}^{p+m}$ the vector of available measurements, we define by the signature matrix associated to z^s, the matrix* $\mathbf{S}$ *given by:*

$$\mathbf{S} = \begin{array}{c} \\ r_{\tau_1} \\ \vdots \\ r_{\tau_p} \end{array} \begin{array}{c} \begin{array}{cccc} \zeta_1 & \zeta_2 & \ldots & \zeta_{p+m} \end{array} \\ \begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} & \ldots & \sigma_{1,p+m} \\ \vdots & \vdots & \ldots & \vdots \\ \sigma_{p,1} & \sigma_{p,2} & \ldots & \sigma_{p,p+m} \end{pmatrix} \end{array} \tag{3.75}$$

with

$$\sigma_{i,j} \triangleq \begin{cases} 0 \quad \text{if} \quad \frac{\partial r_{\tau_i}}{\partial \zeta_j^{(\varrho)}} = 0 \quad \forall \varrho \in \{0, 1, \ldots\} \\ 1 \quad \text{if} \quad \exists\, \varrho \in \{0, 1, \ldots\} \text{ s.t. } \frac{\partial r_{\tau_i}}{\partial \zeta_j^{(\varrho)}} \neq 0 \end{cases} . \tag{3.76}$$

Each column Σ_j of the signature matrix $\mathbf{S}$ indicates whether a residue r_{τ_i} is or is not functionally affected by a fault on the measurement ζ_j. So in (3.76), $\sigma_{i,j} = 0$ means that the residue r_{τ_i} is not affected by a fault on the measurement ζ_j and $\sigma_{i,j} = 1$ means that the residue may be affected.

Definition 3.4.2 (Fault alarm signature) *A column Σ_j of the signature matrix* $\mathbf{S}$ *is called fault alarm signature, associated to the sensor/actuator ζ_j.*

From the signature matrix $\mathbf{S}$ we propose the following definitions of *detectability* and *isolability* in the flatness context:

Definition 3.4.3 (Detectability) *A fault on a sensor/actuator ζ_j is detectable if, and only if there exists at least one $i \in \{1, \ldots, p\}$ such that $\sigma_{i,j} = 1$.*

Definition 3.4.4 (Isolability) *A fault on a sensor S_k, $k = 1, \ldots, p$, is said isolable if, and only if, its corresponding fault alarm signature Σ_k in the signature matrix* $\mathbf{S}$ *is distinct from the others,* i.e.

$$\Sigma_k \neq \Sigma_j, \quad \forall j = 1, \ldots, p+m, \; j \neq k. \tag{3.77}$$

An isolable fault on the actuator A_l, for $l = 1, \ldots, m$, is defined analogously:

$$\Sigma_{p+l} \neq \Sigma_j, \quad \forall j = 1, \ldots, p+m, \; j \neq p+l. \tag{3.78}$$

Definition 3.4.4 indicates that if the signature matrix $\mathbf{S}$ has two identical signatures, *i.e.* $\Sigma_i = \Sigma_j$, for two different sensors/actuators $\zeta_i \neq \zeta_j$, we cannot make a decision on the faulty device. Hence, the fault is detected but cannot be isolated. Accordingly, we introduce the following definition:

Definition 3.4.5 *We define by μ the number of distinct fault alarm signatures of the signature matrix* $\mathbf{S}$ *associated to z^s,* i.e. *μ is the number of isolable faults associated to z^s.*

Therefore, the full isolability of faults is given by the following proposition:

Proposition 3.4.1 *A full isolability of faults is achieved if, and only if, the signature matrix* $\mathbf{S}$ *has $p+m$ distinct fault alarm signatures,* i.e. *$\mu = p+m$.*

In the case where the full isolability is not ensured, *i.e.* $\mu < p+m$, the authors in Martínez-Torres et al. (2014) have proposed to increase the number of residues by using multiple flat outputs. However, we noticed that the choice of these flat outputs is not arbitrary and that they must be *independent* in the sense that when we use them together we gain more isolability of faults. In the next section, we propose a characterization of the flat outputs that are useful for the fault isolation.

3.5 Flat output characterization

In this section, we propose a characterization of the relation between different flat outputs using the notion of *augmented signature matrix*. This characterization leads to a decision concerning the choice of flat outputs that are useful for the isolability of faults.

According to Definition 3.4.5, the number μ of isolable faults by a flat output z is equal to the number of distinct signatures Σ_k of the signature matrix. Then, in order to achieve greater fault isolability, we need to increase the number of distinct fault alarm signatures. This is possible when different projections of the system's output y are available, representing flat outputs (see equation (3.65)).

In the sequel, we denote the i$^{\text{th}}$ element of the set of q flat output vectors Z_i by $Z_i = (z_{i1}, \ldots, z_{im})^T$.

Definition 3.5.1 (Augmented signature matrix) *Let $Z_1, \ldots, Z_q$ be q different flat output vectors of the flat system* (3.62)*, such that $Z_i = \mathrm{pr}_{\mathbb{R}^m}(y^s)$,* i.e. *$Z_i$ is measured by sensors. The* augmented signature matrix $\widetilde{\mathbf{S}}$ *associated to $Z_1, \ldots, Z_q$ is defined by:*

$$\widetilde{\mathbf{S}} = \begin{pmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_q \end{pmatrix} \tag{3.79}$$

where $\mathbf{S}_i$ is the signature matrix associated to the flat output vector Z_i.

Recall that the choice of flat output vectors is not arbitrary. Thus, they must be independent in the sense given by the following definition:

Definition 3.5.2 (Independence) *Let $\widetilde{\mathbf{S}}$ be the augmented signature matrix associated to Z_1 and Z_2:*

$$\widetilde{\mathbf{S}} = \begin{pmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \end{pmatrix},$$

μ_i, $i = 1, 2$, the number of distinct signatures of the matrix $\mathbf{S}_i$ and $\widetilde{\mu}$ the number of distinct signatures of the augmented matrix $\widetilde{\mathbf{S}}$. We say that Z_1 and Z_2 are independent *if, and only if*

$$\widetilde{\mu} > \mu_1 \quad \textit{and} \quad \widetilde{\mu} > \mu_2. \tag{3.80}$$

Definition 3.5.2 indicates that two flat outputs are independent if the number of distinct signatures increases, which corresponds to the number of isolated faults. If the condition (3.80) is not satisfied, then the combination of Z_1 and Z_2 is not helpful for the isolability, and we have to find another combination by calculating more flat outputs. In this context, the condition of full isolability given by Proposition 3.4.1 becomes:

Proposition 3.5.1 *Let $Z_1, \ldots, Z_q$ be q different flat output vectors of the system* (3.62)*. A full isolability of faults on sensors and actuators is achieved if the augmented matrix*

$$\widetilde{\mathbf{S}} = \begin{pmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_q \end{pmatrix} \tag{3.81}$$

has $p + m$ distinct signatures, i.e. *$\widetilde{\mu} = p + m$.*

3.6 Conclusion

The flatness-based fault detection and isolation (FDI) method, presented in this chapter, has been developed to detect and isolate faults on sensors and actuators. The flat output vectors are supposed to be measured by sensors. So, their measurements and their successive time derivatives can be used to calculate the redundant variables. Then, the residue vectors are given by the difference between the real measurements of sensors and actuators and their redundancies.

Furthermore, we have introduced definitions of the detectability and the isolability of faults affecting a system's sensors and actuators, using the notions of signature matrix and fault alarm signature. Sometimes the fault isolability can not be achieved using one flat output vector, *i.e.* there may exist two identical fault alarm signatures for one flat output. For this purpose, the authors of Martínez-Torres et al. (2013a) have proposed to use multiple flat outputs in order to increase the number of residues. The choice of these flat outputs is not arbitrary and they must be independent. In this chapter, we proposed a characterization of the flat outputs that are independent and thus rigorously completed some heuristic results from Martínez-Torres et al. (2013b). Hence, two different flat outputs are independent if, and only if, the number of distinct fault alarm signatures increases in the augmented signature matrix.

In the next chapter, we apply this flatness-based FDI approach to a hydraulic system, the three-tank system. Particularly, we show that by using one flat output, the full isolability of faults cannot be achieved, and thus we need a second flat output. The effectiveness of this approach is demonstrated by simulations and experiments on the real system.

Chapter 4

Application to a Hydraulic System

Abstract

The flatness-based FDI method, presented in the previous chapter, is based on the existence of flat output vector measurements. In this method, the isolability property often requires multiple measured flat outputs to achieve full isolability of faults. This issue leads to a characterization of these flat outputs. That is, two different flat outputs are said to be independent if, by using them together, we obtain more isolability than by using each of them separately. In this chapter, we apply the definitions of the flatness-based FDI method on a hydraulic system, the three-tank system. We show that in order to get full isolability of faults, we need multiple independent flat output vectors. These results are supported by simulations in two cases: open-loop system and closed-loop system. Finally, this method is also applied to an experimental test bench of three tanks.

4.1 Introduction

In this chapter, an application of the flatness-based FDI method, presented in Chapter 3, is performed on a hydraulic system. The three-tank system is a classic example of hydraulic systems, often used in automatic control as a benchmark problem for control reconfiguration and fault diagnosis (Rincon-Pasaye, Martinez-Guerra, and Soria-Lopez, 2008; Rato and Lemos, 1999; Theilliol, Noura, and Ponsart, 2002).

The three-tank system is a multi-input multi-output (MIMO) nonlinear system. It consists of three cylindrical tanks connected to each others by means of pipes, and also connected to a central reservoir. The system is equipped with two pumps that directly supply the two peripheral tanks with water and with three sensors that measure the water level in each tank.

In this study, two cases are considered. The first one is the open-loop system, *i.e.* without any feedback controller. In this case, if a fault occurs on one of the sensors, the actuators will not be affected as these components are not coupled or connected. The second case is the closed-loop system *i.e.* if a fault occurs on one sensor or actuator, the system reacts to compensate this error by means of feedback controllers, and hence the behavior of the system components will also be affected. These two cases differ in the generation of residual signals. Thus, the isolability of faults will also differ in both cases.

Recall that this approach can be applied to detect and isolate multiplicative and additive faults on both sensors and actuators:

- Multiplicative faults: sensor and actuator gains may be reduced from 100% (total measurement) to 0% (complete measurement failure);
- Additive faults: sensors and actuators may represent biases on their measurements.

Therefore, the sum of sensor and actuator faults can be expressed mathematically by (Rao, Xia, and Ying, 2013):

$$\begin{aligned} \mathsf{S}_i^f(t) &= \alpha_i \mathsf{S}_i(t) + \mathsf{S}_{i0} \\ \mathsf{A}_j^f(t) &= \beta_j \mathsf{A}_j(t) + \mathsf{A}_{j0} \end{aligned} \tag{4.1}$$

where $\mathsf{S}_i^f(t)$ and $\mathsf{S}_i(t)$ (resp. $\mathsf{A}_j^f(t)$ and $\mathsf{A}_j(t)$) represent faulty and unfaulty i^{th} sensor (resp. j^{th} actuator) respectively, S_{i0} and A_{j0} are the biases of i^{th} sensor and j^{th} actuator respectively and $0 \leq \alpha_i \leq 1$ and $0 \leq \beta_j \leq 1$ are gain loss factors.

Section 4.2 describes the considered three-tank system and shows its flatness. In Section 4.3, the application of the flatness-based FDI method on the open-loop three-tank system is presented, whereas Section 4.4 is devoted to the closed-loop three-tank system. The results of these two cases are validated by simulations. Moreover, in order to support this methodology of FDI, experimental results are presented in Section 4.5.

4.2 System description

The three-tank system is made up of three cylindrical tanks of cross-sectional area S, connected to each other by means of cylindrical pipes of section S_n, and two pumps P_1 and P_2 that supply tanks T_1 and T_2 with water. These three tanks are also connected to a central reservoir through pipes (see Figure 4.1).

The dynamics of the three-tank system are given by the following nonlinear system:

$$\begin{aligned} \dot{x}_1 &= -Q_{10}(x_1) - Q_{13}(x_1, x_3) + u_1 \\ \dot{x}_2 &= -Q_{20}(x_2) + Q_{32}(x_2, x_3) + u_2 \\ \dot{x}_3 &= Q_{13}(x_1, x_3) - Q_{32}(x_2, x_3) - Q_{30}(x_3) \end{aligned} \tag{4.2}$$

where the state variables x_i, $i = 1, 2, 3$ represent the water level in each tank, Q_{i0}, $i = 1, 2, 3$ is the outflow between each tank and the central reservoir, Q_{13} is the outflow between tanks T_1 and T_3 and Q_{32} the outflow between tanks T_3 and T_2, and u_1 and u_2 are the incoming flows by unit of surface of each pump.

For simplicity's sake, we consider that the valves connecting tanks T_1 and T_3 with the central reservoir are closed, *i.e.* $Q_{10} \equiv 0$ and $Q_{30} \equiv 0$. The expressions of

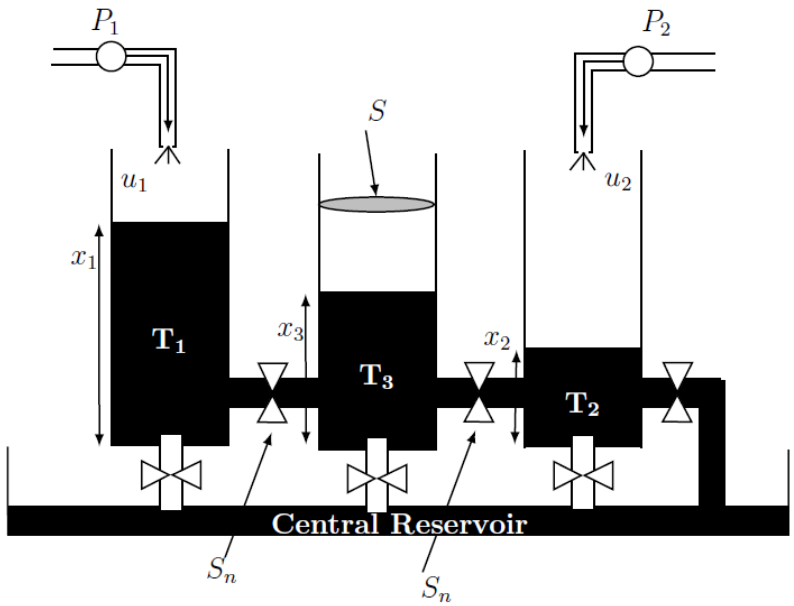

FIGURE 4.1: *Three-Tank System, Source: Noura et al. (2009)*

Q_{13}, Q_{32} and Q_{20} are given by:

$$Q_{13}(x_1, x_3) = \mu_{13}\, sgn\,(x_1 - x_3)\sqrt{|x_1 - x_3|} \tag{4.3}$$

$$Q_{20}(x_2) = \mu_{20}\, sgn\,(x_2)\sqrt{|x_2|} \tag{4.4}$$

$$Q_{32}(x_2, x_3) = \mu_{32}\, sgn\,(x_3 - x_2)\sqrt{|x_3 - x_2|} \tag{4.5}$$

where μ_{13}, μ_{32} and μ_{20} are the flow coefficients.

Each tank T_i is equipped with a sensor S_i to measure its water level x_i. Hence, for this system, the output vector is the same as the state vector:

$$y = (y_1, y_2, y_3)^T = (x_1, x_2, x_3)^T, \tag{4.6}$$

and we denote its measurement by:

$$y^s = (x_1^s, x_2^s, x_3^s)^T. \tag{4.7}$$

The actuators of the system are pumps P_1 and P_2, *i.e.* $\mathsf{A}_1 \triangleq P_1$ and $\mathsf{A}_2 \triangleq P_2$.

4.2.1 Three-tank flat system

Before demonstrating the flatness of the three-tank system, it is important to show that the equilibrium points of this system are first order controllable. If it is not the case, the system will not be flat at these points. So, they must be excluded from the system's configuration (see Remark 2.4.1).

Equilibrium points of the system (4.2) are points $(\widetilde{x}, \widetilde{u})$ such that:

$$\begin{cases} -\mu_{13}\, sgn\,(\widetilde{x}_1 - \widetilde{x}_3)\sqrt{|\widetilde{x}_1 - \widetilde{x}_3|} + \widetilde{u}_1 = 0 \\ -\mu_{20}\, sgn\,(\widetilde{x}_2)\sqrt{|\widetilde{x}_2|} + \mu_{32}\, sgn\,(\widetilde{x}_3 - \widetilde{x}_2)\sqrt{|\widetilde{x}_3 - \widetilde{x}_2|} + \widetilde{u}_2 = 0 \\ \mu_{13}\, sgn\,(\widetilde{x}_1 - \widetilde{x}_3)\sqrt{|\widetilde{x}_1 - \widetilde{x}_3|} - \mu_{32}\, sgn\,(\widetilde{x}_3 - \widetilde{x}_2)\sqrt{|\widetilde{x}_3 - \widetilde{x}_2|} = 0 \end{cases} . \tag{4.8}$$

Then, they are set of points E_1 and E_2 given by:

$$E_1 : \quad \widetilde{x}_1 = \widetilde{x}_2 = \widetilde{x}_3, \quad \widetilde{u}_1 = 0, \quad \widetilde{u}_2 = \mu_{20}\, sgn\,(\widetilde{x}_2)\sqrt{|\widetilde{x}_2|} \tag{4.9}$$

and

$$\begin{aligned} E_2 : \quad & |\widetilde{x}_1 - \widetilde{x}_3| = |\widetilde{x}_3 - \widetilde{x}_2|, \quad \widetilde{u}_1 = \mu_{13}\, sgn\,(\widetilde{x}_1 - \widetilde{x}_3)\sqrt{|\widetilde{x}_1 - \widetilde{x}_3|}, \\ & \widetilde{u}_2 = \mu_{20}\, sgn\,(\widetilde{x}_2)\sqrt{|\widetilde{x}_2|} - \mu_{32}\, sgn\,(\widetilde{x}_3 - \widetilde{x}_2)\sqrt{|\widetilde{x}_3 - \widetilde{x}_2|}. \end{aligned} \tag{4.10}$$

It is easy to show that the equilibrium point E_2 is first order controllable, whereas the equilibrium point E_1 is not. Thus, to avoid intrinsic singularities that may affect the flatness of the system, the following configuration is considered:

$$(\mathcal{C}) : \quad x_1 > x_3 > x_2 > 0. \tag{4.11}$$

The system (4.2) under the configuration (4.11) is flat with $z = (x_1, x_3)^T = (z_1, z_2)^T$ as flat output. In fact, the state vector $x = (x_1, x_2, x_3)^T$ can be expressed in

function of z and its successive time derivatives as follows:

$$\begin{aligned} x_1 &= z_1 \\ x_2 &= z_2 - \frac{1}{\mu_{32}^2}\Big(- \dot{z}_2 + \mu_{13}\sqrt{z_1 - z_2}\Big)^2 \\ x_3 &= z_2. \end{aligned} \tag{4.12}$$

Inputs u_1 and u_2 can also be expressed in function of z and its successive time derivatives using the first two equations of (4.2):

$$\begin{aligned} u_1 &= \dot{z}_1 + \mu_{13}\sqrt{z_1 - z_2} \\ u_2 &= \dot{x}_2 + \mu_{20}\sqrt{x_2} - \mu_{32}\sqrt{z_2 - x_2} \end{aligned} \tag{4.13}$$

with x_2 given by (4.12). The flat output z can also be calculated using the Smith decomposition or the direct representation of the unimodular completion algorithm (see Proposition 3.2.5).

4.2.2 Reference trajectory of the system

In the next simulations on the three-tank system, the reference trajectories are generated by flatness, using the flat output $z = (x_1, x_3)^T$ (see Section 2.5). The initial and final conditions for the flat output are given by:

$$\begin{aligned} x_{1_i} = 0.15\ m, \quad x_{1_f} = 0.3\ m \\ x_{3_i} = 0.1\ m, \quad x_{3_f} = 0.25\ m. \end{aligned} \tag{4.14}$$

The initial and final time-points are respectively $t_i = 50s$ and $t_f = 350s$. The duration of the simulations is of $t = 400s$. The reference trajectory $t \mapsto z_{ref}(t)$, associated to the flat output $z = (x_1, x_3)^T$, is calculated using a fifth order polynomial interpolation (see (2.53)). Therefore, reference trajectories of the system inputs and outputs are deduced from (4.12) and (4.13). Figure 4.2 represents the reference trajectory for each tank T_i, and Figure 4.3 represents reference trajectories for the inputs u_1 and u_2.

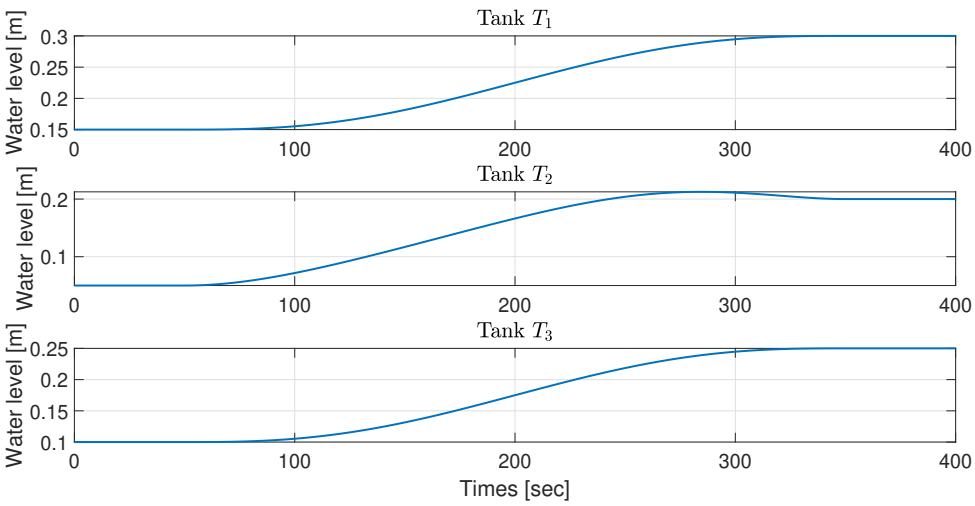

FIGURE 4.2: *Reference trajectories of the water level in each tank T_i*

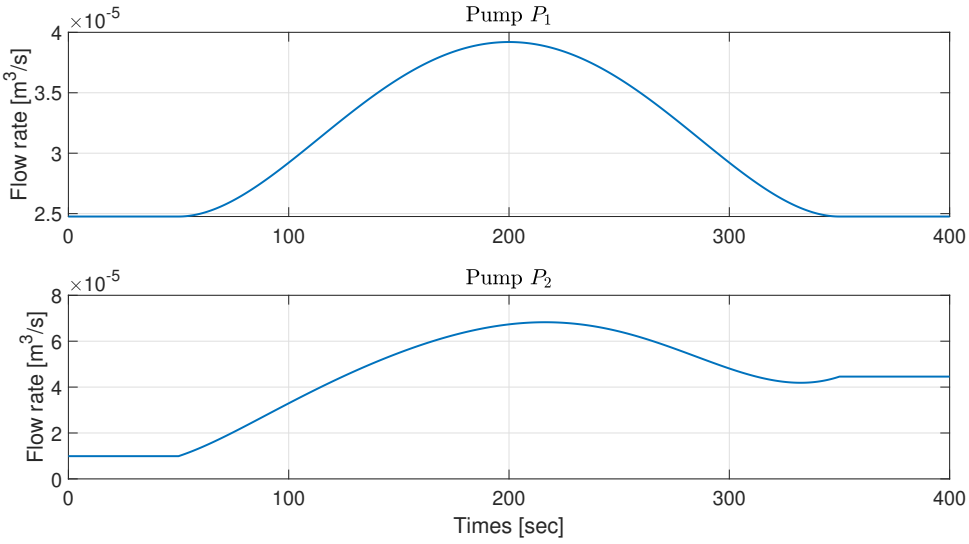

FIGURE 4.3: *Reference trajectories of the flow rate of each Pump P_i*

In the following two sections, definitions that are given in Chapter 3 are applied on the three-tank system for the aim of FDI. Two cases are considered: the open-loop system and the closed-loop system. Moreover, we will show that a single flat output vector is not sufficient to isolate all possible faults on the system. Then, we use a supplementary flat output to get full isolability.

4.3 FDI process in an open-loop system

An open-loop system, also referred to as non-feedback system, is a type of continuous control systems in which the output has no influence or effect on the control action of the input signal. In other words, in an open-loop control system the output can be measured but not "fed back" for comparison with the input. Therefore, an open-loop system is expected to faithfully follow its input control or set point regardless of the final result. So, in the case of fault on one sensor, this fault does not affect the other components.

4.3.1 Case A: one flat output

Consider the flat output $z = (x_1, x_3)^T = (z_1, z_2)^T$. It is a projection of the output vector $y = (x_1, x_2, x_3)^T$ on $\mathbb{R}^2$ (see (3.65)). Thus, the components z_1 and z_2 are measured by sensors S_1 and S_3, respectively. In the following, we denote by $z^s = (x_1^s, x_3^s)^T = (z_1^s, z_2^s)^T$ the measured flat output.

In order to construct the vector of residues, first we compute the redundant inputs and outputs, using (4.12) and (4.13):

$$\begin{aligned}
x_1^z &= z_1^s \\
x_2^z &= z_2^s - \frac{1}{\mu_{32}^2}\Big(- \dot{z}_2^s + \mu_{13}\sqrt{z_1^s - z_2^s}\Big)^2 \\
x_3^z &= z_2^s \\
u_1^z &= \dot{z}_1^s + \mu_{13}\sqrt{z_1^s - z_2^s} \\
u_2^z &= \dot{x}_2^z + \mu_{20}\sqrt{x_2^z} - \mu_{32}\sqrt{z_2^s - x_2^z}.
\end{aligned} \tag{4.15}$$

Figures 4.4, 4.5 and 4.6 show the redundant signals x_2^z, u_1^z and u_2^z, computed in the fault-free case. Since x_1^s and x_3^s are the components of z^s, their associated redundant signals $x_1^z = x_1^s$ and $x_3^z = x_3^s$.

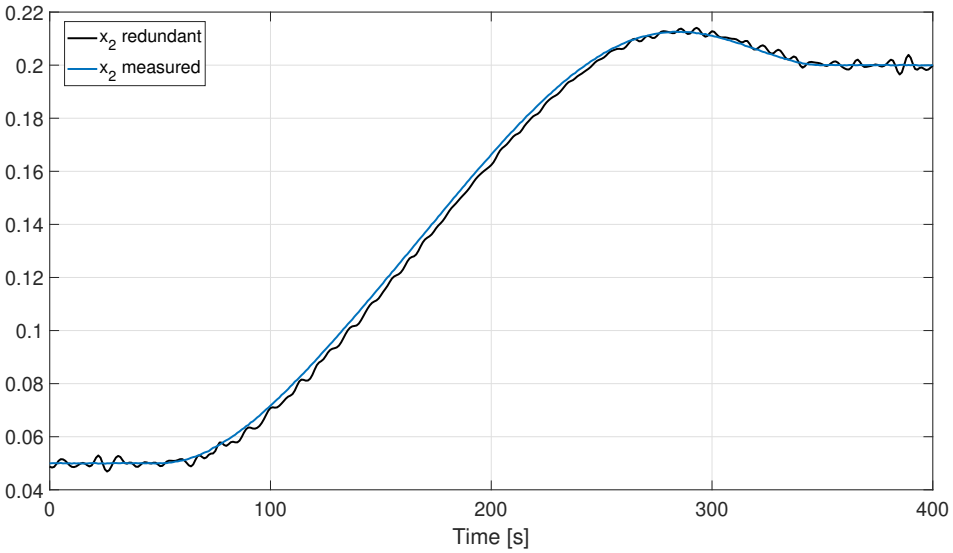

FIGURE 4.4: *Redundant signal x_2^z, calculated using the measured flat output z, vs. the measured output x_2^s*

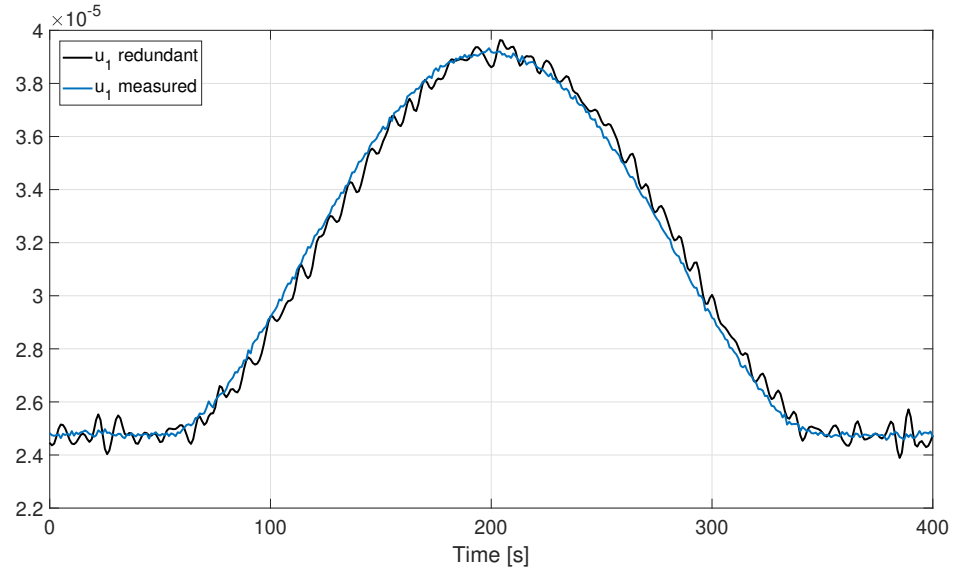

FIGURE 4.5: *Redundant signal u_1^z, calculated using the measured flat output z, vs. the input signal u_1*

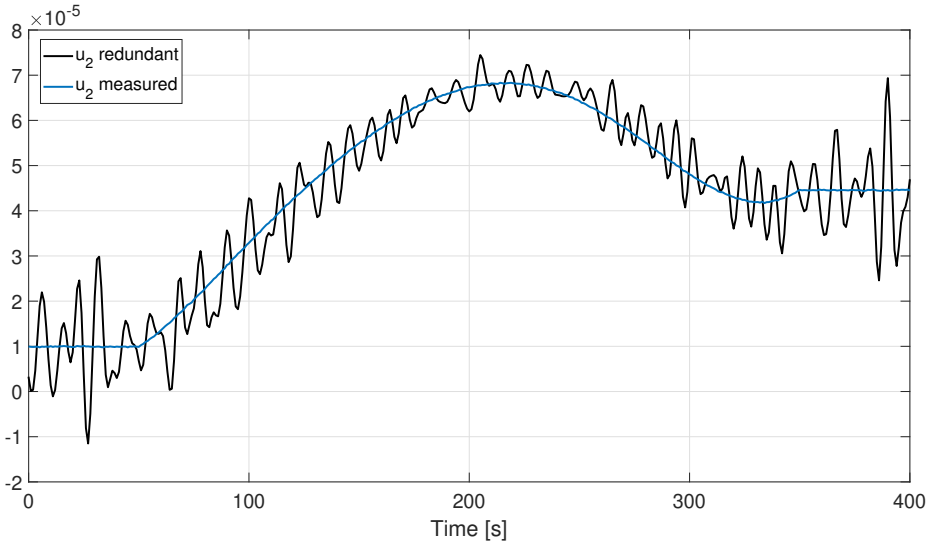

FIGURE 4.6: *Redundant signal u_2^z, calculated using the measured flat output z, vs. the input signal u_2*

According to Definition 3.3.1, the vector of residues, associated to z^s, is given by:

$$r = \begin{pmatrix} R_{S_1} \\ R_{S_2} \\ R_{S_3} \\ R_{A_1} \\ R_{A_2} \end{pmatrix} = \begin{pmatrix} x_1^s \\ x_2^s \\ x_3^s \\ u_1 \\ u_2 \end{pmatrix} - \begin{pmatrix} x_1^z \\ x_2^z \\ x_3^z \\ u_1^z \\ u_2^z \end{pmatrix}. \tag{4.16}$$

However, residues R_{S_1} and R_{S_3} are identically zero:

$$\begin{aligned} R_{S_1} &= x_1^s - x_1^z = z_1^s - z_1^s = 0 \\ R_{S_3} &= x_3^s - x_3^z = z_2^s - z_2^s = 0 \end{aligned} \tag{4.17}$$

thus, according to (3.74), the vector r is truncated to:

$$r_\tau = \begin{pmatrix} r_{\tau_1} \\ r_{\tau_2} \\ r_{\tau_3} \end{pmatrix} = \begin{pmatrix} R_{\mathsf{S}_2} \\ R_{\mathsf{A}_1} \\ R_{\mathsf{A}_2} \end{pmatrix}. \tag{4.18}$$

The vector ζ of measured variables, introduced in Definition 3.4.1, is given by:

$$\zeta = (x_1^s, x_2^s, x_3^s, u_1, u_2) \in \mathbb{R}^5. \tag{4.19}$$

Therefore, the signature matrix $\mathbf{S}$, associated to z^s, is of dimension 3×5:

$$\mathbf{S} = \begin{array}{c} \\ r_{\tau_1} \\ r_{\tau_2} \\ r_{\tau_3} \end{array} \begin{array}{c} \begin{array}{ccccc} x_1^s & x_2^s & x_3^s & u_1 & u_2 \end{array} \\ \begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} & \sigma_{1,3} & \sigma_{1,4} & \sigma_{1,5} \\ \sigma_{2,1} & \sigma_{2,2} & \sigma_{2,3} & \sigma_{2,4} & \sigma_{2,5} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma_{3,3} & \sigma_{3,4} & \sigma_{3,5} \end{pmatrix} \end{array} \tag{4.20}$$

and constructed as follows:

– All the residues in (4.18) depend on the measurement of $z^s = (x_1^s, x_3^s)^T$ and its successive time derivatives (see (4.15)), then the first and the third columns of the signature matrix contain only ones:

$$\sigma_{i,1} = \sigma_{i,3} = 1, \qquad \forall i = 1, 2, 3.$$

– Only residue r_{τ_1} depends on x_2^s, then the second column will be such that:

$$\sigma_{1,2} = 1 \quad \text{and} \quad \sigma_{i,2} = 0,\ i = 2, 3.$$

– Since r_{τ_2} depends only on u_1 and r_{τ_3} depends only on u_2, then column 4 and column 5 of $\mathbf{S}$ are such that:

$$\sigma_{2,4} = 1 \quad \text{and} \quad \sigma_{i,4} = 0\ \forall i = 1, \ldots, 3,\ i \neq 2$$

and

$$\sigma_{3,5} = 1 \quad \text{and} \quad \sigma_{i,5} = 0 \; \forall i = 1, \ldots, 3, \; i \neq 3$$

respectively.

Hence, the signature matrix (4.20) becomes:

$$\mathbf{S} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}. \tag{4.21}$$

According to Definition 3.4.3, all faults on the system sensors and actuators are detectable. In addition, since fault alarm signatures Σ_2, Σ_4 and Σ_5 are distinct, faults on sensor S_2 and actuators A_1 and A_2 are isolable, according to Definition 3.4.4. This reflects the fact that if, at some point during system operation, a fault alarm is launched with the signature Σ_2 then we conclude that the sensor S_2 is faulty. Therefore, the number of isolable faults by the flat output z^s is

$$\mu = 3. \tag{4.22}$$

However, if we obtain a signature like Σ_1, the fault could be on the sensor S_1 or S_3, since signatures Σ_1 and Σ_3 are identical. Then, a fault on S_1 or S_3 cannot be isolated. These results are confirmed by simulations for both additive and multiplicative faults.

Simulation results

In our simulations, we consider, for multiplicative faults, a 20% failure for sensors and actuators: at time $t = 200s$, the sensors measure 80% of the actual water level measurements instead of 100%, and for actuators, a 20% failure is considered. For additive faults, a bias of $+0.1m$ is considered for sensors, and an extra flow of

$10^{-5}m^3/s$ is considered for actuators at time $t = 200s$. Parameters of the three-tank system, used in our simulations, are given in Table 4.1.

Parameters	Symbol	Value
Tank sectional area	S	$0.0154\ m^2$
Pipes sectional area	S_n	$5 \times 10^{-5}\ m^2$
Outflow coefficients	μ_{13}, μ_{32}	1.107×10^{-4}
Outflow coefficient	μ_{20}	1.55×10^{-4}
Maximum water level	h_{max}	$0.62\ m$
Maximum flow rate	u_{max}	$10^{-4}\ m^3/s$

TABLE 4.1: *Parameter values of the three-tank system*

Threshold setting and derivative estimation

White Gaussian noise is added to the measured outputs and inputs of the system with a level corresponding to the actual process level (see Table 4.2). In order to fix a threshold for each residue, several fault-free simulations were realized with different initial and final conditions. The amplitude of the detection threshold is calculated by selecting the worst case among all the simulation results, plus a 5% safety margin to avoid false alarms caused by measurement noise or modeling errors. The values of the maximum and minimum threshold for each residue are given in Table 4.3.

	Mean	Variance	Power
Sensor	0	0.13	2×10^{-4}
Actuator	0	0.13	2.3804×10^{-7}

TABLE 4.2: *Parameter values of the added white Gaussian noise*

Moreover, due to the presence of noise on the sensors, derivatives of the flat output measurements may not exist. Thus, as presented in Section 3.2, these derivatives have to be estimated. In our simulations, we use a *Butterworth Low-pass Filter* of order $N = 4$ and cutoff frequency $f_c = 0.25Hz$, to filter noise components. Then,

	Max	Min
$R_{S_2}\ [m]$	5.9085×10^{-4}	-8.665×10^{-4}
$R_{A_1}[m^3/s]$	2.1686×10^{-7}	-2.3535×10^{-7}
$R_{A_2}[m^3/s]$	3.2452×10^{-6}	-2.2423×10^{-6}

TABLE 4.3: *Case A: values of the maximum and minimum threshold for each residue*

derivatives are calculated using the *Discrete Filtered Derivative* represented by the following transfer function (Group et al., 1992):

$$\left(\frac{K}{T}\right)\frac{z-1}{z+Ts/T-1} \tag{4.23}$$

where $K = 1$ is the gain, $T = 20s$ is the time constant and $Ts = 1s$ is the sample time.

Simulations have been realized within a period of $400s$ and faults have been applied at $t = 200s$. We recall that only one fault affects the sensors and actuators at a time. In the following the thresholds of the Table 4.3 are normalized between -1 and 1.

Multiplicative faults

For multiplicative faults, the sensor sensitivity drops from 100% to 80% at $t = 200s$, *i.e.* the sensor loses 20% of its measurement, and the actuator provides only 80% of its work capability. Figure 4.7 shows that if a fault appears on the sensor S_2 only the residue R_{S_2} exceeds its threshold which corresponds to the signature Σ_2 of the signature matrix **S** given by (4.21):

$$\Sigma_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{4.24}$$

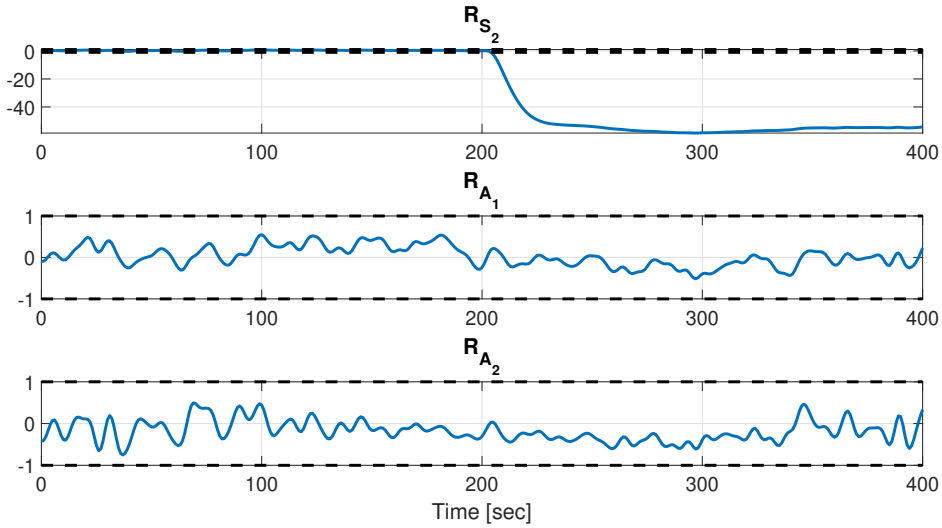

FIGURE 4.7: *Case A: multiplicative fault on sensor* S_2 *at time* $t = 200s$ *with normalized thresholds*

Similarly, a fault on pump P_1 affects only the residue R_{A_1} (see Figure 4.8), and a fault on pump P_2 affects only the residue R_{A_2} (see Figure 4.9), which corresponds to signatures Σ_4 and Σ_5 respectively:

$$\Sigma_4 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \Sigma_5 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{4.25}$$

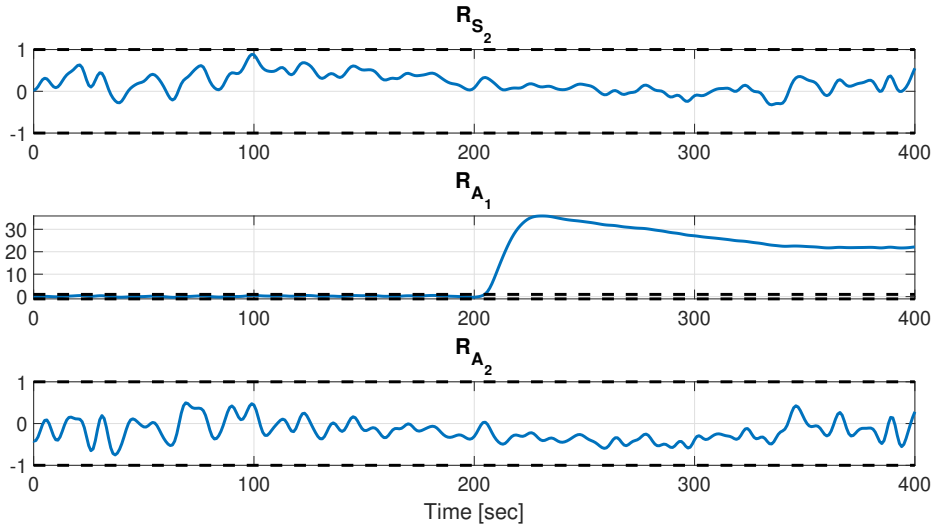

FIGURE 4.8: *Case A: multiplicative fault on actuator* A_1 *at time* $t = 200s$ *with normalized thresholds*

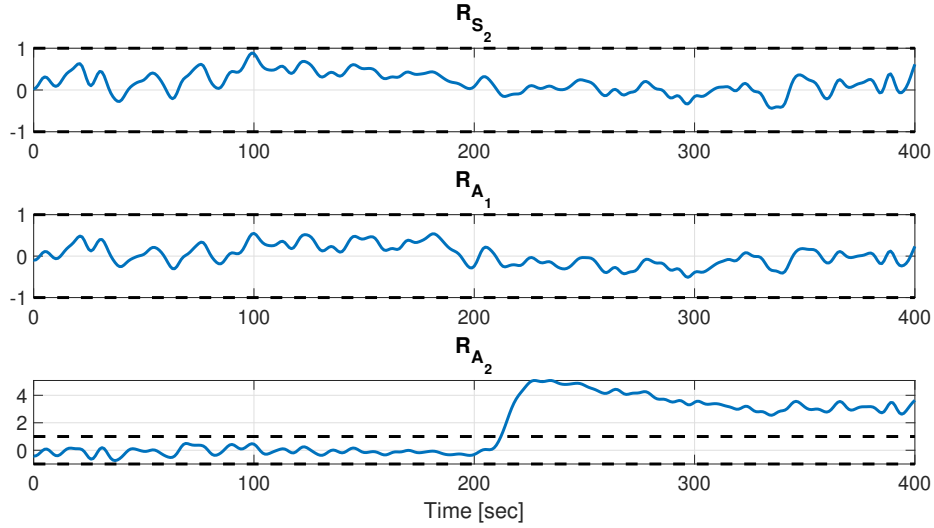

FIGURE 4.9: *Case A: multiplicative fault on actuator* A_2 *at time* $t = 200s$ *with normalized thresholds*

Simulations also show that if a fault affects sensor S_1 or S_3, whose measurements are parts of the flat output $z^s = (x_1^s, x_3^s)^T$, then all the residues will be affected which

corresponds to signatures Σ_1 and Σ_3:

$$\Sigma_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \Sigma_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \tag{4.26}$$

See Figure 4.10 and Figure 4.11.

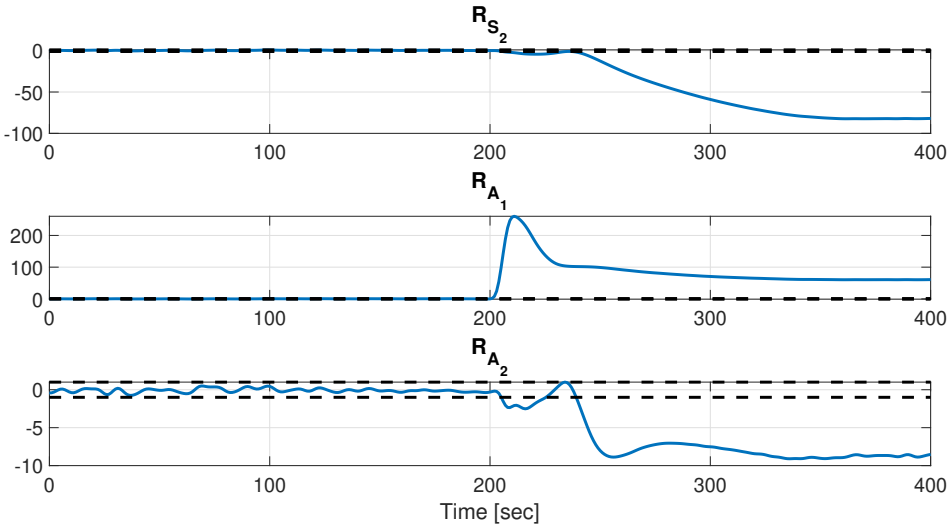

FIGURE 4.10: *Case A: multiplicative fault on sensor* S_1 *at time* $t = 200s$ *with normalized thresholds*

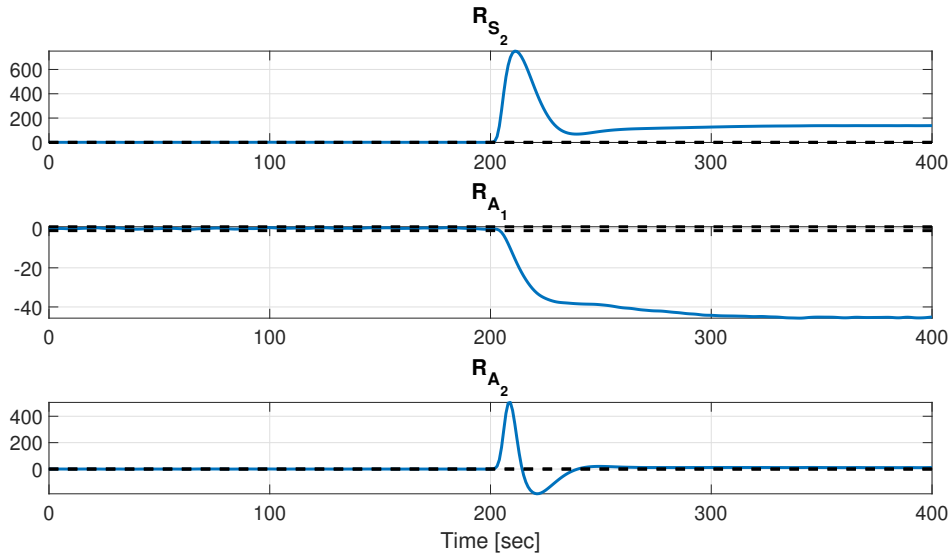

FIGURE 4.11: *Case A: multiplicative fault on sensor* S_3 *at time* $t = 200s$ *with normalized thresholds*

Additive faults

For additive faults at time $t = 200s$, a bias of $+0.1m$ is added to each sensor, and an extra flow of $10^{-5}m^3/s$ is added to each pump. Figures 4.12, 4.13, 4.14, 4.15 and 4.16 illustrate that additive faults on sensors and actuators give the same signatures as for multiplicative faults. Then the signature matrix for additive and multiplicative faults is identical to the signature matrix theoretically given by (4.21):

$$\mathbf{S} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}. \tag{4.27}$$

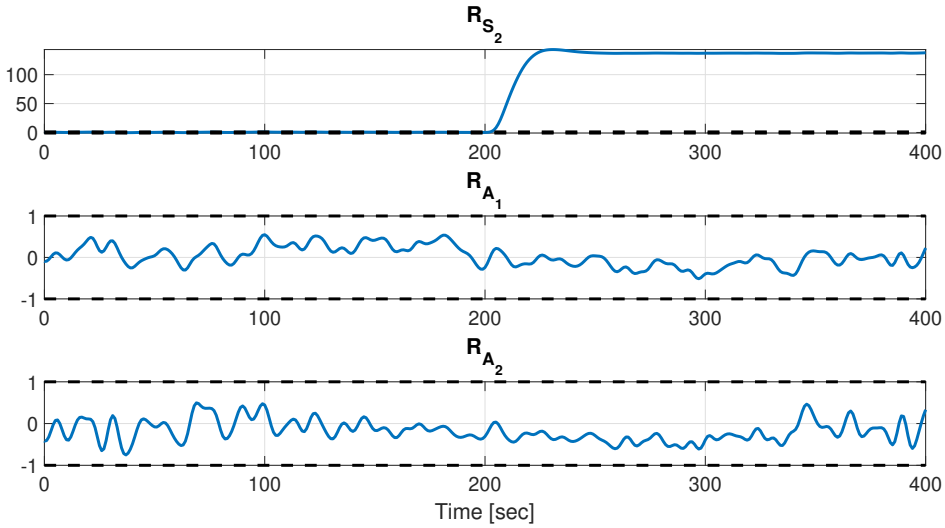

FIGURE 4.12: *Case A: additive fault on sensor* S_2 *at time* $t = 200s$ *with normalized thresholds*

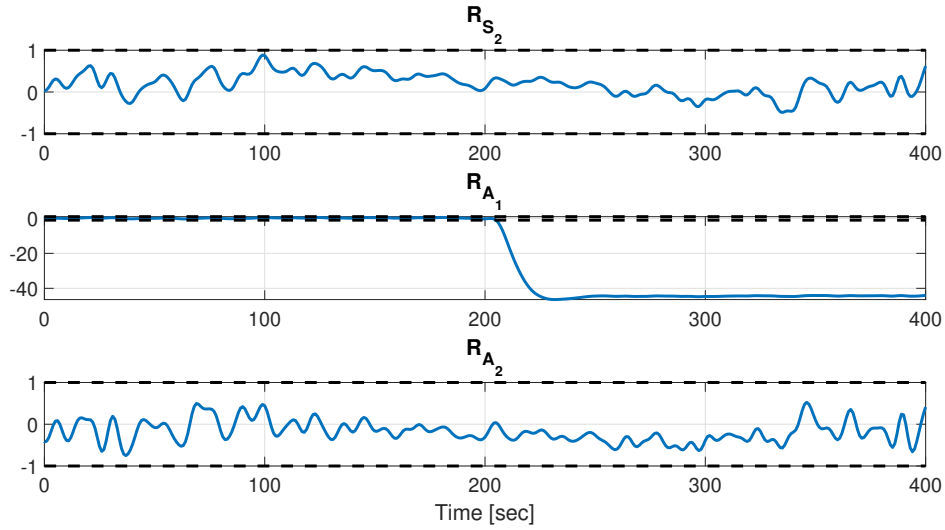

FIGURE 4.13: *Case A: additive fault on actuator* A_1 *at time* $t = 200s$ *with normalized thresholds*

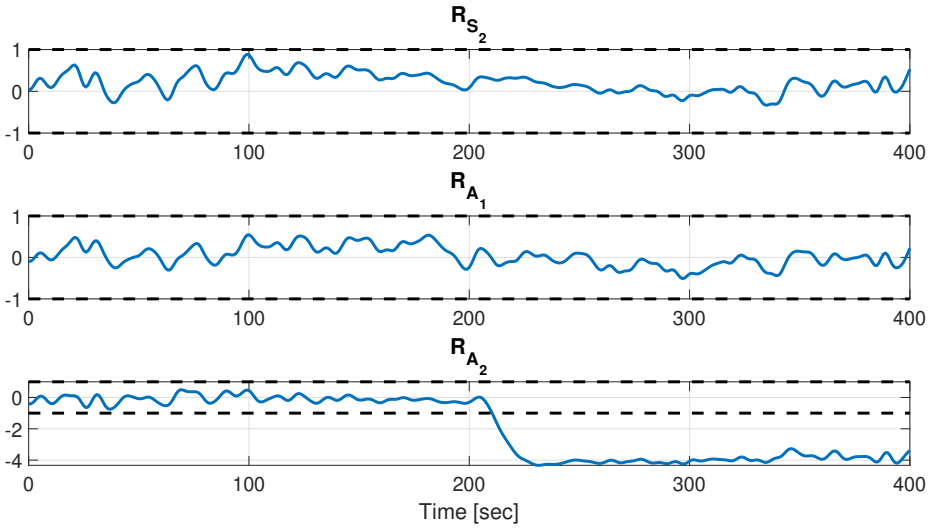

FIGURE 4.14: *Case A: additive fault on actuator* A_2 *at time* $t = 200s$ *with normalized thresholds*

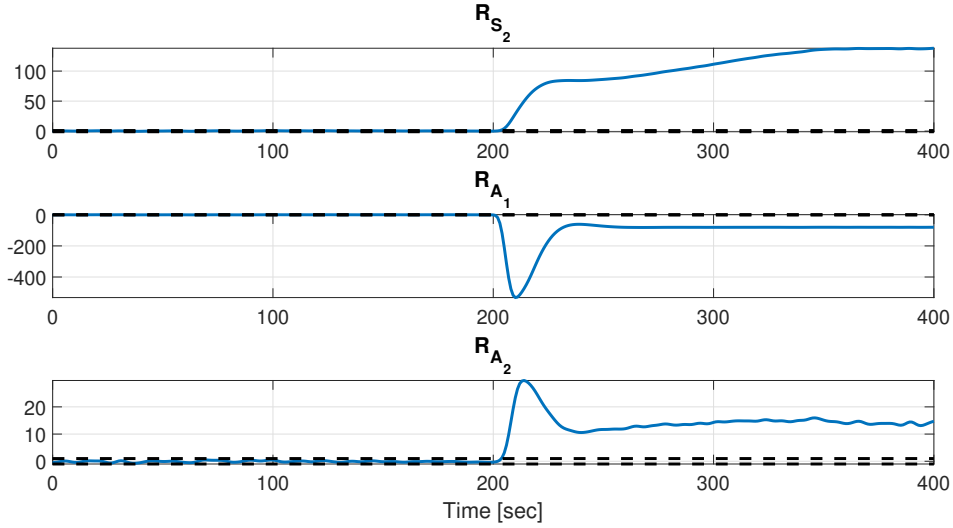

FIGURE 4.15: *Case A: additive fault on sensor* S_1 *at time* $t = 200s$ *with normalized thresholds*

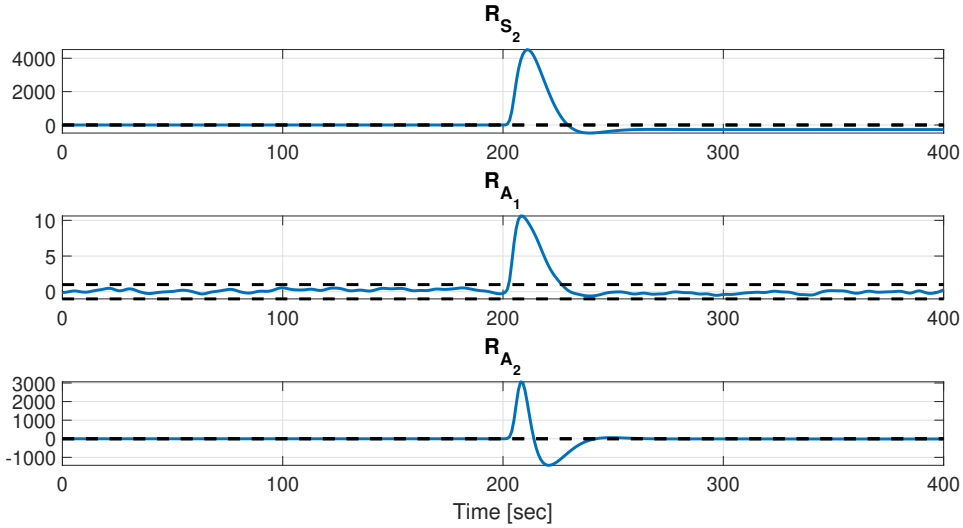

FIGURE 4.16: *Case A: additive fault on sensor* S_3 *at time* $t = 200s$ *with normalized thresholds*

In this case, the full isolability of faults is not achieved, *i.e.* the number of distinct fault alarm signatures is $\mu = 3 < p + m$. Thus, a second flat output vector is needed to ensure the full isolability.

4.3.2 Case B: two flat outputs

In the following, we denote by Z_1 the flat output vector $Z_1 = (z_{11}^s, z_{12}^s)^T = (x_1^s, x_3^s)^T$. The corresponding vector of residues is given by (4.18). We recall the signature matrix associated to Z_1, and we denote it by $\mathbf{S}_1$:

$$\mathbf{S}_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}. \tag{4.28}$$

We also recall that faults on sensors S_1 and S_3 cannot be isolated. The number of distinct signatures of $\mathbf{S}_1$ is $\mu_1 = 3$.

In order to increase the number of isolable faults, we have to find another flat output vector that is independent from Z_1, in the sense given by Definition 3.5.2.

We consider another flat output vector $Z_2 = (z_{21}, z_{22})^T = (x_2, x_3)^T$, measured by sensors S_2 and S_3, *i.e.* $Z_2 = (z^s_{21}, z^s_{22})^T = (x^s_2, x^s_3)^T$.

To construct the vector of residues associated to Z_2 and its signature matrix, we calculate first the redundant inputs and outputs using (3.67) and (3.68):

$$\begin{aligned}
x_1^{Z_2} &= z^s_{22} + \frac{1}{\mu_{13}^2}\left(\dot{z}^s_{22} + \mu_{32}\sqrt{z^s_{22} - z^s_{21}}\right)^2 \\
x_2^{Z_2} &= z^s_{21} \\
x_3^{Z_2} &= z^s_{22} \\
u_1^{Z_2} &= \dot{x}_1^{Z_2} + \mu_{13}\sqrt{x_1^{Z_2} - z^s_{22}} \\
u_2^{Z_2} &= \dot{z}^s_{21} + \mu_{20}\sqrt{z^s_{21}} - \mu_{32}\sqrt{z^s_{22} - z^s_{21}}.
\end{aligned} \tag{4.29}$$

Therefore, as shown for the flat output Z_1, residues $R^{Z_2}_{S_2}$ and $R^{Z_2}_{S_3}$ are identically zero and the truncated vector of residues reads:

$$r^{Z_2}_{\tau} = \begin{pmatrix} R^{Z_2}_{S_1} \\ R^{Z_2}_{A_1} \\ R^{Z_2}_{A_2} \end{pmatrix} = \begin{pmatrix} x^s_2 \\ u_1 \\ u_2 \end{pmatrix} - \begin{pmatrix} x_2^{Z_2} \\ u_1^{Z_2} \\ u_2^{Z_2} \end{pmatrix}. \tag{4.30}$$

All residues in (4.30) depend on the measurements x^s_2 and x^s_3 of sensors S_2 and S_3, hence all the residues are affected by a fault on S_2 or S_3, *i.e.*

$$\Sigma_2^{Z_2} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad \text{and} \quad \Sigma_3^{Z_2} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}. \tag{4.31}$$

Only the residue $R^{Z_2}_{\mathsf{S}_1}$ depends on the measurement of S_1, hence only this residue will be affected by a fault on S_1, *i.e.*

$$\Sigma_1^{Z_2} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{4.32}$$

Similarly, only $R^{Z_2}_{\mathsf{A}_1}$ depends on the measurement of A_1 and only $R^{Z_2}_{\mathsf{A}_2}$ depends on the measurement of A_2:

$$\Sigma_4^{Z_2} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \Sigma_5^{Z_2} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{4.33}$$

Hence, the signature matrix associated to Z_2 is given by:

$$\mathbf{S}_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}. \tag{4.34}$$

Signatures Σ_1, Σ_4 and Σ_5 in the matrix $\mathbf{S}_2$ are distinct, then, according to Definition 3.4.4, faults on sensor S_1 and actuators A_1 and A_2 are isolable by the flat output Z_2. Moreover, the number of distinct signatures of $\mathbf{S}_2$ is $\mu_2 = 3$. However, since signatures Σ_2 and Σ_3 are identical, then faults on sensors S_2 and S_3 cannot be isolated.

It remains to be verified whether the two flat outputs Z_1 and Z_2 are independent. The augmented signature matrix associated to Z_1 and Z_2 is given by:

$$\widetilde{\mathbf{S}} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}. \tag{4.35}$$

The number of distinct fault alarm signatures of $\widetilde{\mathbf{S}}$ is $\widetilde{\mu} = 5$, and we have

$$\widetilde{\mu} > \mu_1 \quad \text{and} \quad \widetilde{\mu} > \mu_2. \tag{4.36}$$

Then, according to Definition 3.5.2, the flat output vectors Z_1 and Z_2 are independent. Moreover, since $\widetilde{\mu} = p + m$, the flat output vectors Z_1 and Z_2 ensure full isolability of faults on the three-tank system.

Simulation Results

In simulations, multiplicative faults are added to sensors and actuators independently at time $t = 200s$. The values of the thresholds are given in Table 4.4. These thresholds are then normalized between -1 and 1.

Figure 4.17 shows that only the residues $R^{Z_1}_{A_1}$ and $R^{Z_1}_{A_2}$, which are independent of the measurement x^s_2 of the sensor S_2, are not affected by the fault on S_2, which explains the fault alarm signature Σ_2 of $\widetilde{\mathbf{S}}$.

	Max	Min
$R^{Z_1}_{S_2}\ [m]$	5.9085×10^{-4}	-8.665×10^{-4}
$R^{Z_1}_{A_1}\ [m^3/s]$	2.1686×10^{-7}	-2.3535×10^{-7}
$R^{Z_1}_{A_2}\ [m^3/s]$	3.2452×10^{-6}	-2.2423×10^{-6}
$R^{Z_2}_{S_1}\ [m]$	6.8399×10^{-4}	-1×10^{-3}
$R^{Z_2}_{A_1}\ [m^3/s]$	2.7607×10^{-6}	-3.0339×10^{-6}
$R^{Z_2}_{A_2}\ [m^3/s]$	3.5951×10^{-7}	-2.9841×10^{-7}

TABLE 4.4: *Case B: values of the maximum and minimum threshold for each residue*

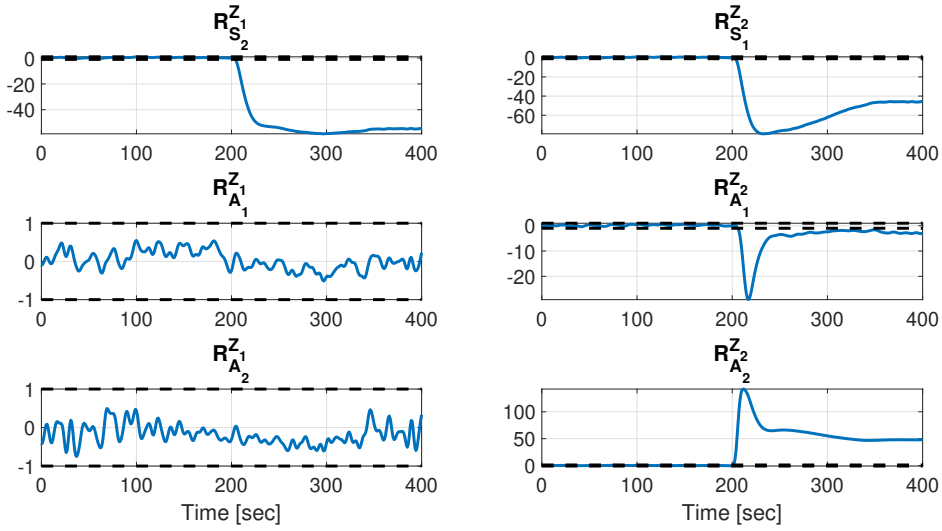

FIGURE 4.17: *Case B: multiplicative fault on sensor* S_2 *at time* $t = 200s$ *with normalized thresholds*

Figure 4.18 shows that only residues depending on the measurement u_1 exceed their thresholds when a fault affects the actuator A_1. We obtain the same behavior if a fault affects the actuator A_2 (see Figure 4.19).

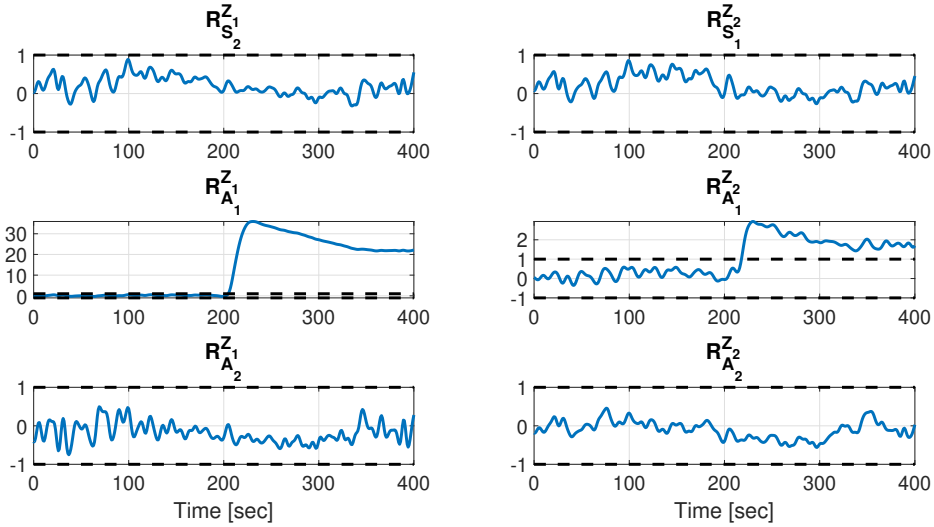

FIGURE 4.18: *Case B: multiplicative fault on actuator* A_1 *at time* $t = 200s$ *with normalized thresholds*

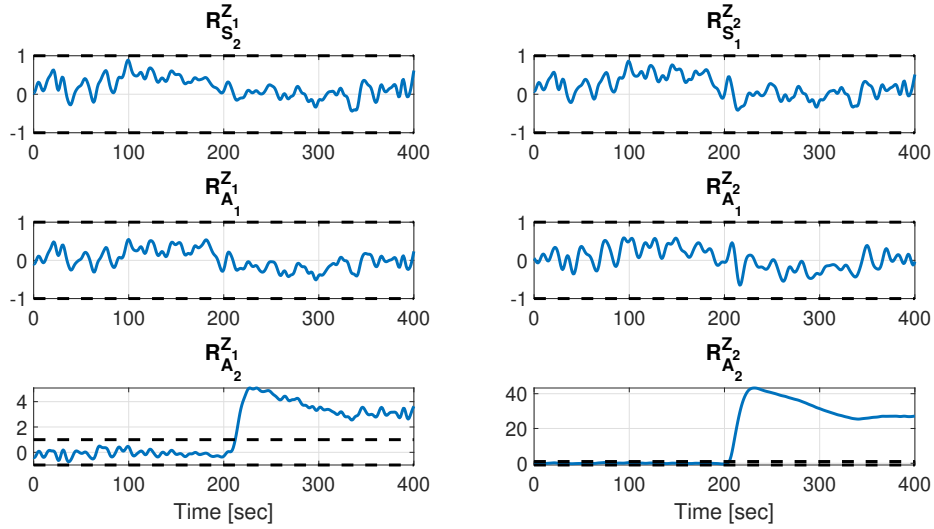

FIGURE 4.19: *Case B: multiplicative fault on actuator* A_2 *at time* $t = 200s$ *with normalized thresholds*

As the flat outputs Z_1 and Z_2 are independent, faults that affect sensor S_1 or S_3 are now isolable, contrary to the results of the section 4.3.1. A fault that affects sensor S_1 does not affect residues $R^{Z_2}_{A_1}$ and $R^{Z_2}_{A_2}$ (see Figure 4.20), and a fault that affects sensor S_3 affects all the residues (see Figure 4.21). Additive faults act in the same way.

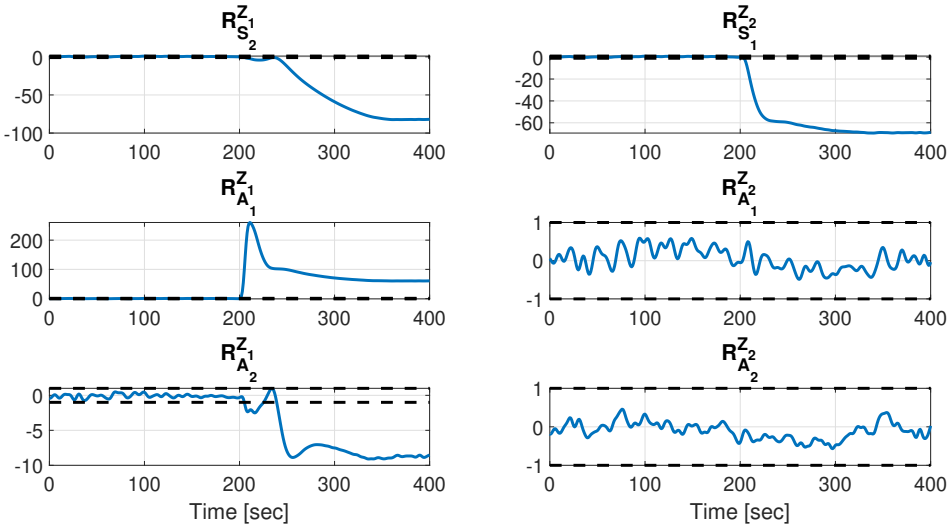

FIGURE 4.20: *Case B: multiplicative fault on sensor* S_1 *at time* $t = 200s$ *with normalized thresholds*

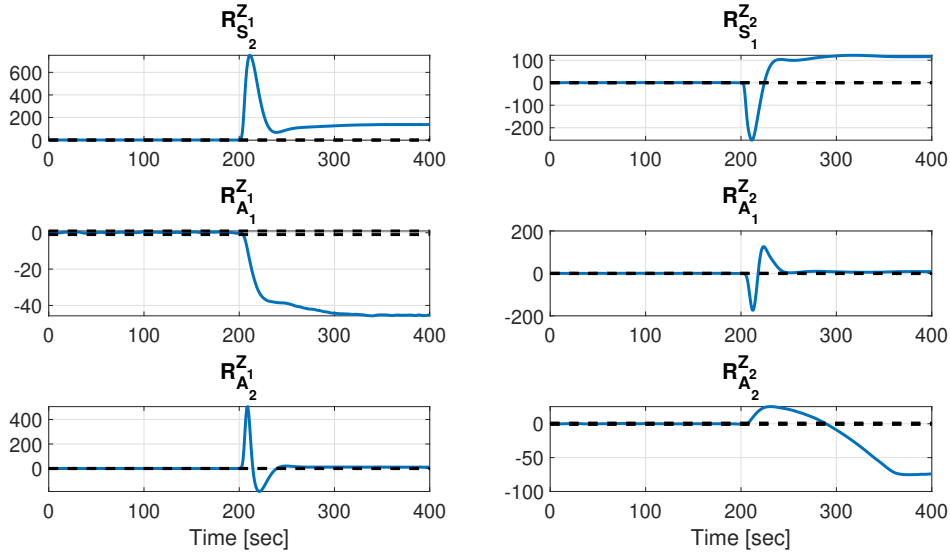

FIGURE 4.21: *Case B: multiplicative fault on sensor* S_3 *at time* $t = 200s$ *with normalized thresholds*

4.4 FDI process in a closed-loop system

A closed-loop control system, also known as a feedback control system, is a system which uses the concept of an open-loop system as its forward path but has one or more feedback loops or paths between its output and its input. It is a fully automatic control system in which control action depends on the output in some way.

4.4.1 The behavior of the system against failures

The fault in a closed-loop system is propagated with the feedback loop, increasing the difficulty of fault isolation. Recall that the measured output vector is given by:

$$y^s = (y_1^s, \ldots, y_p^s), \tag{4.37}$$

and that the flat output is also measured by sensors, *i.e.* $z^s = \text{pr}_{\mathbb{R}^m}(y^s)$. We also suppose that the system is controlled by a set of m controllers C_j, for $j = 1, \ldots, m$. Then, the input u, in this case, can be expressed, using the Laplace transform, as follows:

$$U_l(s) = U_l^{ref}(s) + C_l(s)(Y_l^{ref}(s) - Y_l^s(s)), \quad \text{for} \quad l = 1, \ldots, m \tag{4.38}$$

where $U_l^{ref}(s)$ and $Y_l^{ref}(s)$ are the reference input and reference output, respectively and s is the Laplace variable. Therefore, the l^{th}-input residue R_{A_l}, for $l = 1, \ldots, m$, of Definition 3.3.1 becomes:

$$R_{\mathsf{A}_l} = u_l - u_l^z \tag{4.39}$$

where u_l is replaced by (4.38). Then a fault that appears on sensor S_l affects directly the actuator A_l.

Back to the three-tank system, we introduce the following PI controllers:

$$\text{PI}_1 \quad \triangleq \quad K_1 = 0.001, \quad \tau_1 = 11.5 \tag{4.40}$$

$$\text{PI}_2 \quad \triangleq \quad K_2 = 0.001, \quad \tau_2 = 12.5, \tag{4.41}$$

in order to control the water level in tanks T_1 and T_2 (Martínez-Torres et al., 2014). Then, a fault that affects sensors S_1 or S_2 is propagated with the feedback and affects pumps P_1 and P_2, respectively. For example, suppose that the sensor S_1 loses 20% of its measurement. The system reacts as there is a loss of water in tank T_1. Then, pump P_1 works more to cover this loss, see Figure 4.23/Pump 1. Therefore, the water level in tank T_1 increases until the measurement of the sensor reaches its reference level (see Figure 4.22/Tank 1), which also leads to an elevation on the real water level of the tank. As a result, the water level in tank T_3 increases as

well and exceeds its reference trajectory (see Figure 4.22/Tank 3). Finally, the water level rises slightly in the tank T_2 which reduces the pressure on the pump P_2 using controller PI_2 (see Figure 4.23/Pump 2).

The system reacts in the same way if a fault affects the sensor S_2. However, this time the system reacts as there is a loss of water in tank T_2 and then pump P_2 puts in more work to cover this loss in the sensor, which increases the real water level in the tank. Therefore, the water level will also increases in tank T_3 and a little less in the tank T_1, which forces pump P_1 to reduce its pressure.

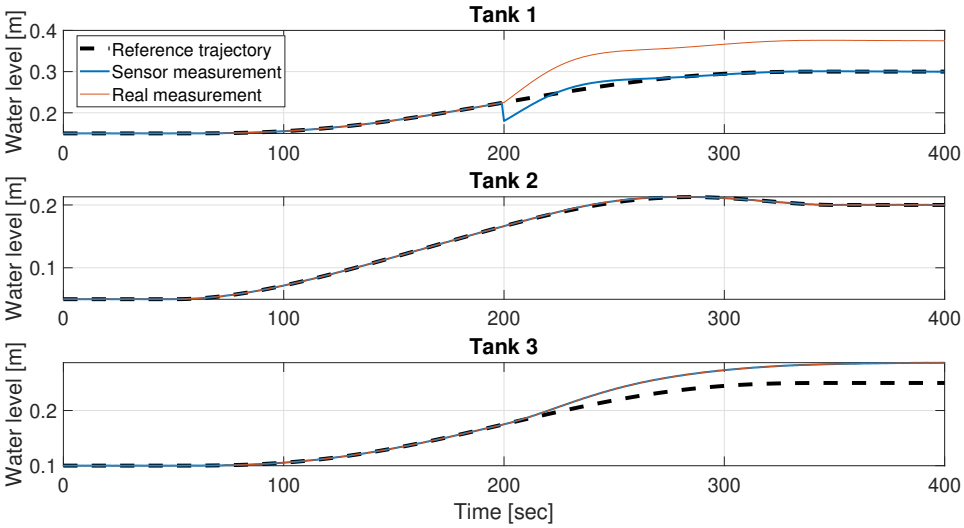

FIGURE 4.22: *Water level in each tank against multiplicative fault on sensor* S_1 *at time* $t = 200s$

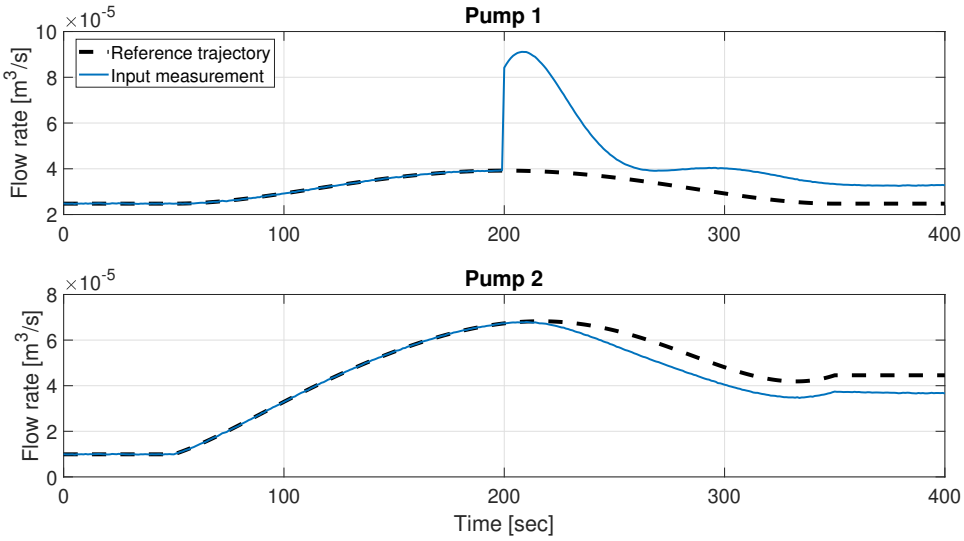

FIGURE 4.23: *Flow rate of each pump against multiplicative fault on sensor* S_1 *at time* $t = 200s$

However, the water level in tank T_3 is not controlled, then the sensor S_3 is not coupled with pump P_1 or P_2. Thus, if sensor S_3 loses 20% of its measurement, the other components will no be affected, see Figures 4.24 and 4.25. The system reacts as if it were an open-loop system.

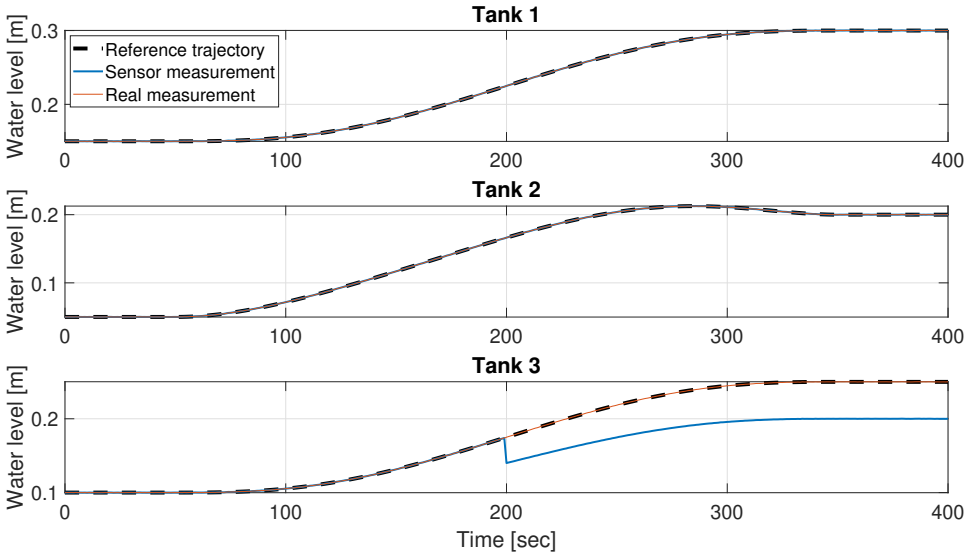

FIGURE 4.24: *Water level in each tank against multiplicative fault on sensor* S_3 *at time* $t = 200s$

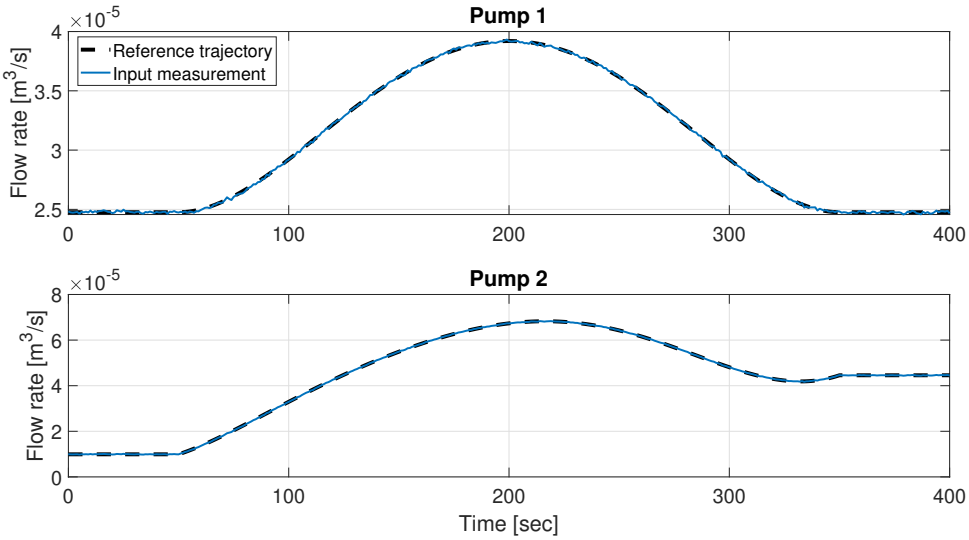

FIGURE 4.25: *Flow rate of each pump against multiplicative fault on sensor* S_3 *at time* $t = 200s$

In addition, a fault that affects an actuator also affects the system states and outputs. For example, suppose that pump P_1 provides only 80% of the desired input u_1^{ref} at time $t = 200s$, then the water level in tank T_1 decreases, and the controller PI_1 reacts to compensate the fault. Figures 4.26 and 4.27 illustrate the behavior of the system against the multiplicative fault on pump P_1. The same will happen if the pump P_2 is faulty, this time the controller PI_2 will react to compensate the fault.

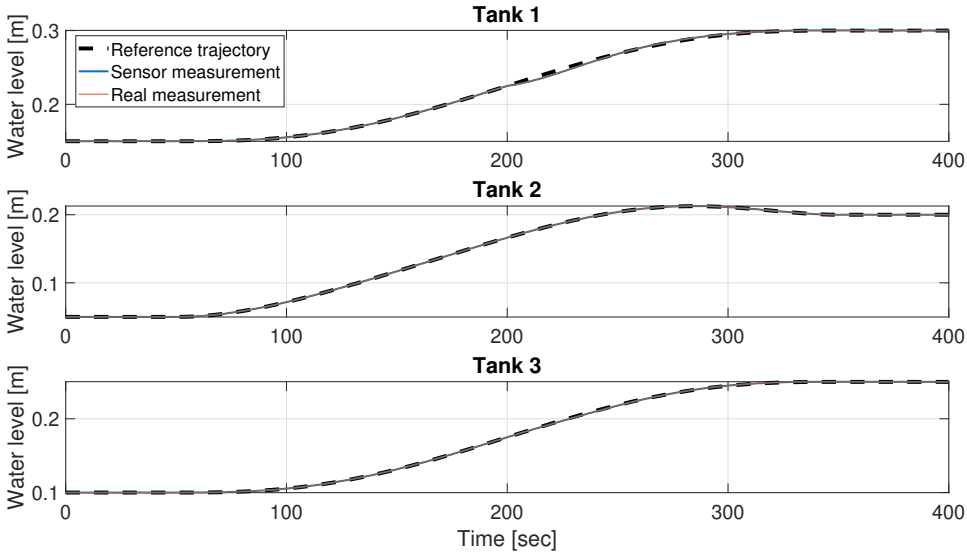

FIGURE 4.26: *Water level in each tank against multiplicative fault on pump* P_1 *at time* $t = 200s$

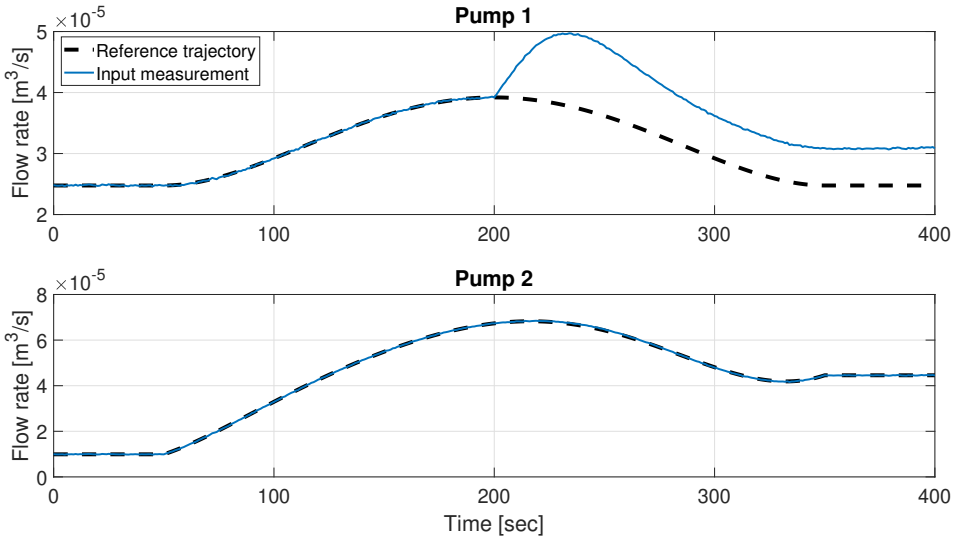

FIGURE 4.27: *Flow rate of each pump against multiplicative fault on pump P_1 at time $t = 200s$*

Additive faults react in the same way as multiplicative faults. In the rest of the chapter, we will apply only multiplicative faults for simulations and experiments.

4.4.2 Fault detection and isolation

The FDI in the case of a closed-loop system is the same as in the case of an open-loop system, presented in Chapter 3. The only difference is in the input of the system: in the open-loop system $U_l(s) = U_l^{ref}(s)$, for $l = 1, \ldots, m$, whereas, in the closed-loop system $U_l(s) = U_l^{ref}(s) + C_l(s)(Y_l^{ref}(s) - Y_l^s(s))$, for $l = 1, \ldots, m$ (see (4.38)).

For the example of the three-tank system, sensor S_1 is related to pump P_1 by means of controller PI_1, then

$$U_1(s) = U_1^{ref}(s) + PI_1(s)(Y_1^{ref}(s) - Y_1^s(s)). \tag{4.42}$$

Similarly, sensor S_2 is related to pump P_2 by means of controller PI_2:

$$U_2(s) = U_2^{ref}(s) + PI_2(s)(Y_2^{ref}(s) - Y_2^s(s)). \tag{4.43}$$

In section 4.3, we have shown that the three-tank system is flat with $Z_1 = (x_1^s, x_3^s)^T$ and $Z_2 = (x_2^s, x_3^s)^T$ two flat outputs. Then, the redundant inputs and outputs are calculated in function of Z_1 and Z_2 and their successive time derivatives

(see (4.15) and (4.29)). The residue vector, in the closed-loop case, is then given by:

$$r = \begin{pmatrix} R^{Z_1}_{\mathsf{S}_2} \\ R^{Z_1}_{\mathsf{A}_1} \\ R^{Z_1}_{\mathsf{A}_2} \\ R^{Z_2}_{\mathsf{S}_1} \\ R^{Z_2}_{\mathsf{A}_1} \\ R^{Z_2}_{\mathsf{A}_2} \end{pmatrix} = \begin{pmatrix} y_2^s - y_2^z \\ u_1 - u_1^z \\ u_2 - u_2^z \\ y_1^s - y_1^z \\ u_1 - u_1^z \\ u_2 - u_2^z \end{pmatrix} \tag{4.44}$$

with u_1 and u_2 given by (4.42) and (4.43), respectively.

Theoretically, according to Definition 3.3.1, the augmented signature matrix $\widetilde{S}$, associated to Z_1 and Z_2 is given by:

$$\widetilde{S} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}. \tag{4.45}$$

This augmented signature matrix has also been verified by simulations.

4.4.2.1 Multiplicative faults on sensors

Thresholds for the residues in the closed-loop case are given in Table 4.5.

We start by applying a fault on sensor S_1: at time $t = 200s$, the sensor measures only 80% of the actual water level. Figure 4.28 shows that all the residues that depend on the measurement of S_1 exceed their thresholds: $R^{Z_1}_{\mathsf{S}_2}$, $R^{Z_1}_{\mathsf{A}_1}$, $R^{Z_1}_{\mathsf{A}_2}$, $R^{Z_2}_{\mathsf{S}_1}$ and

	Max	Min
$R_{S_2}^{Z_1}\ [m]$	5.9027×10^{-4}	-8.6457×10^{-4}
$R_{A_1}^{Z_1}\ [m^3/s]$	$2.2.006 \times 10^{-7}$	-2.3912×10^{-7}
$R_{A_2}^{Z_1}\ [m^3/s]$	3.2452×10^{-6}	-2.2512×10^{-6}
$R_{S_1}^{Z_2}\ [m]$	6.8257×10^{-4}	-2.835×10^{-3}
$R_{A_1}^{Z_2}\ [m^3/s]$	4.3244×10^{-6}	-3.0342×10^{-6}
$R_{A_2}^{Z_2}\ [m^3/s]$	7.2568×10^{-7}	-3.0158×10^{-7}

TABLE 4.5: *Values of the maximum and minimum threshold for each residue in the closed-loop case*

$R_{A_1}^{Z_2}$. The fault alarm signature Σ_1, associated to the sensor S_1, is then given by:

$$\Sigma_1 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}^T . \tag{4.46}$$

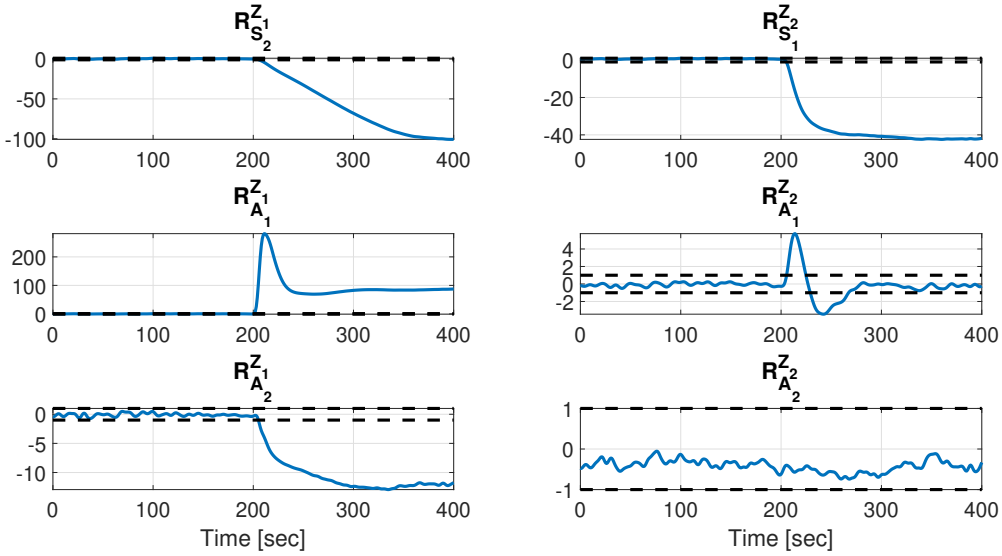

FIGURE 4.28: *Closed-loop system: multiplicative fault on sensor S_1 at time $t = 200s$*

Remark 4.4.1 *Figure 4.29 shows the measurement of the sensor S_1 against the real water level in the tank T_1 and the redundant variable x_1^z, calculated by flatness. For the aim of fault tolerant control (FTC), the redundant variable x_1^z can replace the faulty sensor, and*

then the system returns to its normal state. For more details on flatness-based fault tolerant control, see Martínez-Torres et al. (2014).

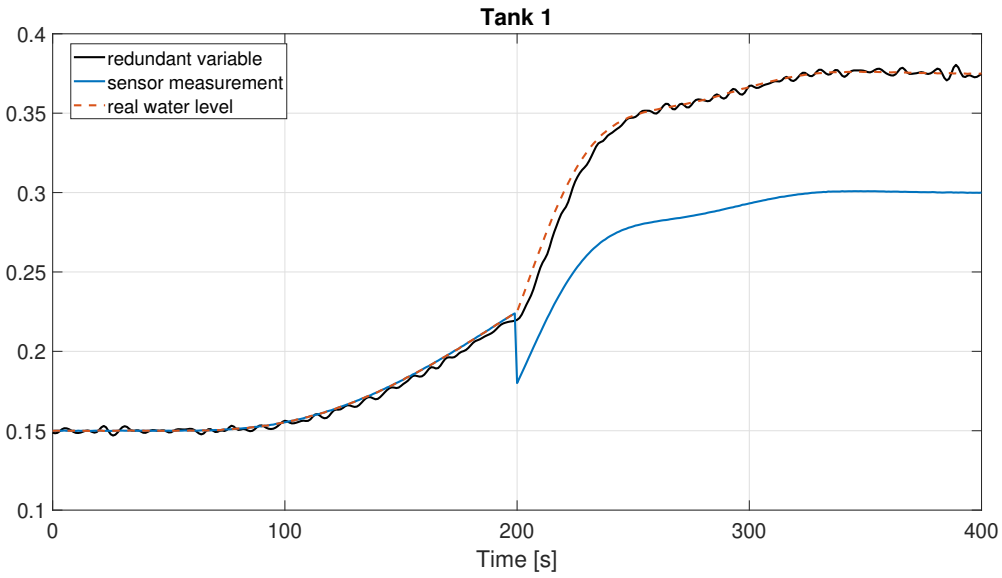

FIGURE 4.29: *Closed-loop system: fault reconfiguration after a multiplicative fault on sensor* S_1 *at time* $t = 200s$

Consider the same fault on the sensor S_2. Again, all the residues that depend on the measurement y_2^s of S_2 exceed their thresholds: $R_{S_2}^{Z_1}$, $R_{S_1}^{Z_2}$, $R_{A_1}^{Z_2}$ and $R_{A_2}^{Z_2}$. Additionally, since the fault on sensor S_2 affects the pump P_2, the residue $R_{A_2}^{Z_1}$ is triggered temporarily. The fault alarm signature Σ_2 becomes in this case:

$$\Sigma_2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}^T \tag{4.47}$$

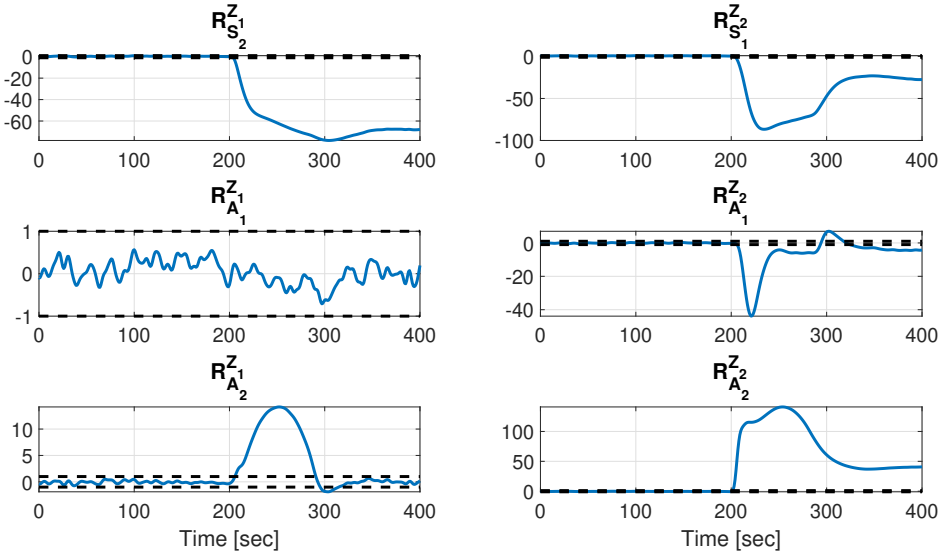

FIGURE 4.30: *Closed-loop system: multiplicative fault on sensor* S_2 *at time* $t = 200s$

Since the measurement of sensor S_3 is part of the flat outputs Z_1 and Z_2, all the residues exceed their thresholds in case a fault occurs on the sensor S_3, see Figure 4.31. The fault alarm signature Σ_3 is then given by:

$$\Sigma_3 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}^T. \tag{4.48}$$

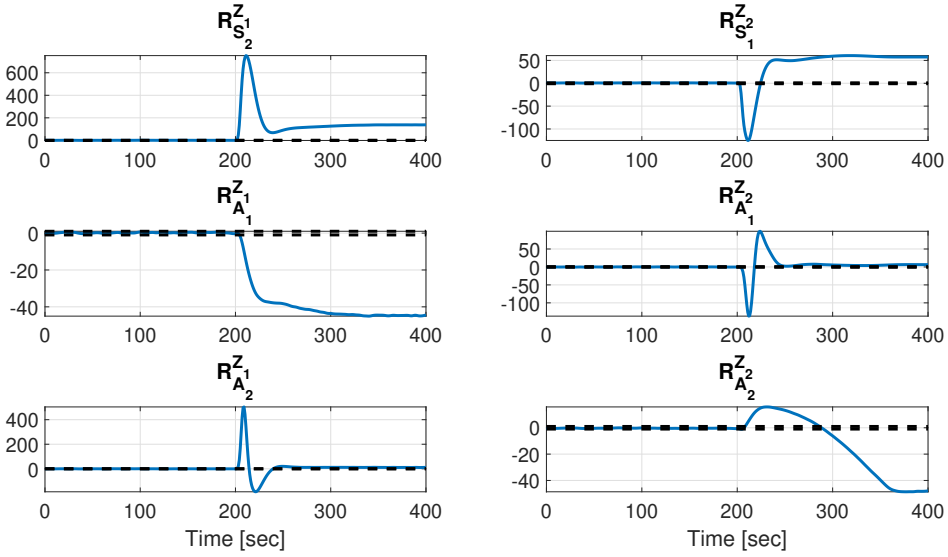

FIGURE 4.31: *Closed-loop system: multiplicative fault on sensor* S_3 *at time* $t = 200s$

4.4.2.2 Multiplicative faults on actuators

Consider a multiplicative fault on the pump P_1. The input measurement u_1 of P_1 is not part of the flat output vectors Z_1 and Z_2. Then, only the residues $R^{Z_1}_{A_1}$ and $R^{Z_2}_{A_1}$ are affected by a fault on P_1, see Figure 4.32. The fault alarm signature Σ_4 is then given by:

$$\Sigma_4 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}^T . \tag{4.49}$$

A multiplicative fault on the pump P_2 reacts in the same way (see Figure 4.33). Its associated fault alarm signature is then given by:

$$\Sigma_5 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}^T . \tag{4.50}$$

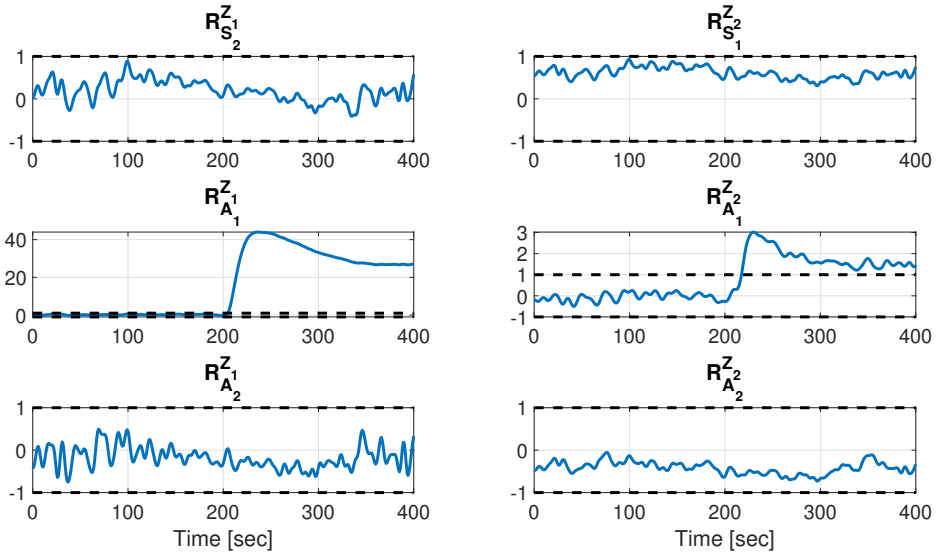

FIGURE 4.32: *Closed-loop system: multiplicative fault on pump P_1 at time $t = 200s$*

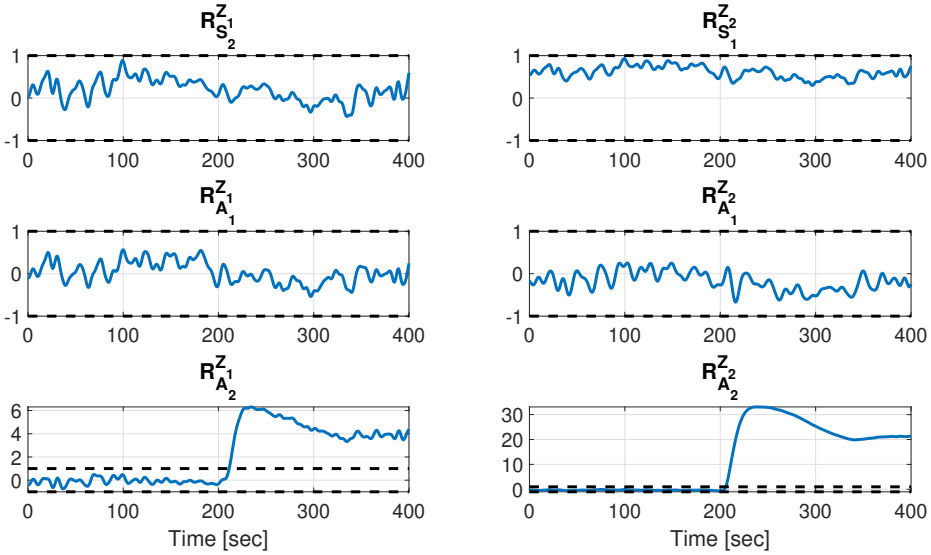

FIGURE 4.33: *Closed-loop system: multiplicative fault on pump P_2 at time $t = 200s$*

Finally, according to Proposition 3.5.1, the number of distinct fault alarm signatures in $\widetilde{\mathbf{S}}$ is equal to $p + m = 5$, then by using the two flat outputs Z_1 and Z_2, in the closed-loop case, we still have full isolability of faults.

Additive faults on sensors and actuators react in the same way giving the augmented signature matrix $\widetilde{\mathbf{S}}$.

4.5 Experimental results

In order to improve the effectiveness of the flatness-based FDI method presented in Chapter 3, we perform several experiments on the real DTS200 three-tank system.

The DTS200 three-tank system is a popular laboratory equipment which is considered to be an interesting experimental system for the study of nonlinear control system as well as FTC and FDI. All the experiments in this thesis have been performed on the system of three-tank presented in Figure 4.34.

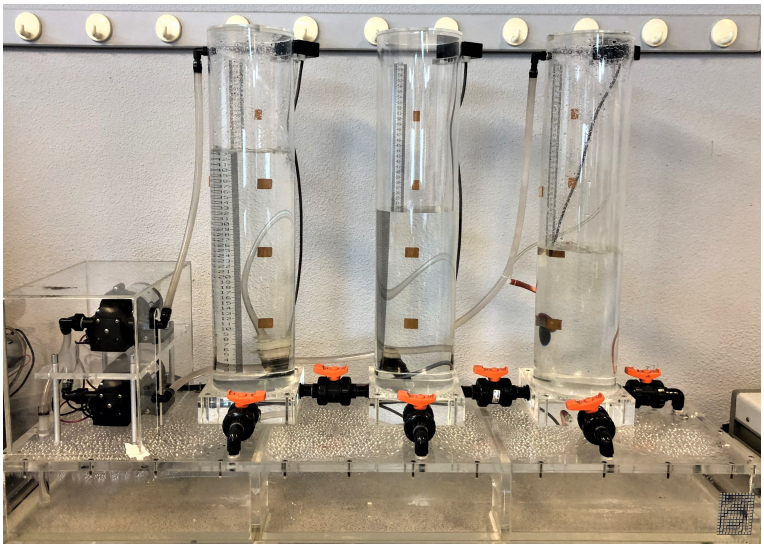

FIGURE 4.34: *DTS200 three-tank System.*

The parameters of the DTS200 three-tank system are given in Table 4.6.

Parameters	Symbol	Value
Tank sectional area	S	$0.0154\ m^2$
Pipes sectional area	S_n	$5 \times 10^{-5}\ m^2$
Outflow coefficient	μ_{13}	8.5273×10^{-5}
Outflow coefficient	μ_{32}	8.5563×10^{-5}
Outflow coefficient	μ_{20}	1.5901×10^{-4}
Maximum water level	h_{max}	$0.62\ m$
Maximum flow rate	u_{max}	$10^{-4}\ m^3/s$

TABLE 4.6: *Parameter values of the real three-tank system*

The DTS200 three-tank model is connected to a PC with the help of a PC plug-in card MF624. For the aim of FDI, the DTS200 three-tank model provides the application of any type of fault on both sensors and actuators:

- Multiplicative faults: sensor and actuator gains may be reduced from 100% (total measurement) to 0% (complete measurement failure) by the mean of potentiometers;
- Additive faults: sensors and actuators may represent biases on their measurements.

4.5.1 Reference trajectory and control of the system

Same as in section 4.2.2, reference trajectories of the water level $x_i(t)$, $i = 1, 2, 3$, in each tank and of the outflow u_j, $j = 1, 2$, of each pump are calculated using the flatness property: consider $z = (x_1, x_3)$ the flat output, we construct a reference trajectory $t \to z_{ref}(t)$ for z using a fifth order polynomial interpolation. The initial and final conditions of the flat output, considered in our experiments, are:

$$\begin{aligned} x_{1_i} &= 0.2m, \quad x_{1_f} = 0.35m \\ x_{3_i} &= 0.15m, \quad x_{3_f} = 0.25m \end{aligned} \tag{4.51}$$

at initial time $t_i = 50s$ and final time $t_f = 350s$. The duration of each experiment is $t = 400s$. The reference trajectories of the states and the inputs of the system are then deduced by flatness using (2.50) (see Section 2.5).

In order to control the system against external faults and disturbances, we use the same PI controllers as in simulations:

$$\mathrm{PI}_1 \quad \triangleq \quad K_1 = 0.001, \quad \tau_1 = 11.5 \tag{4.52}$$

$$\mathrm{PI}_2 \quad \triangleq \quad K_2 = 0.001, \quad \tau_2 = 12.5. \tag{4.53}$$

The controller PI_1 is linked to tank T_1 while the controller PI_2 is linked to tank T_2. Reference trajectories and the real measurements of the water levels and outflows are illustrated in Figures 4.35 and 4.36.

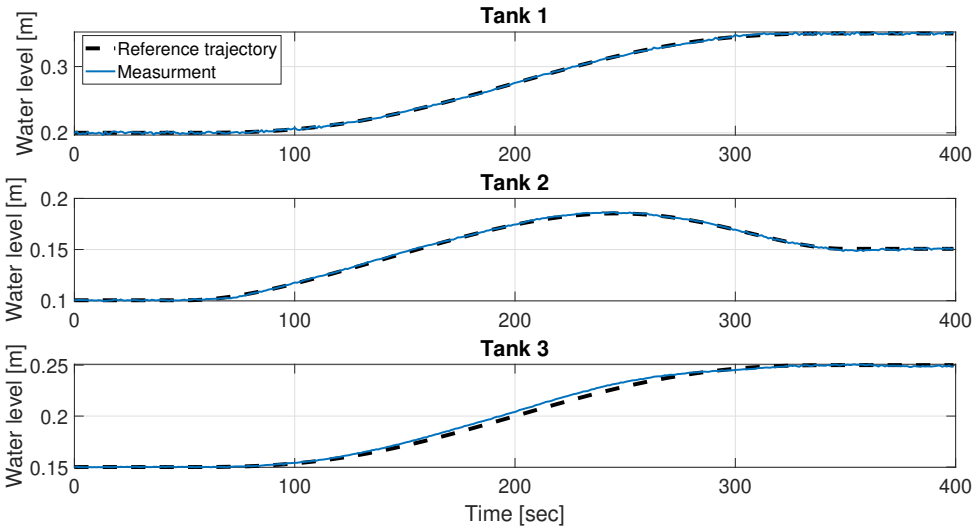

FIGURE 4.35: *Reference trajectories vs. measurements of the water level in each tank*

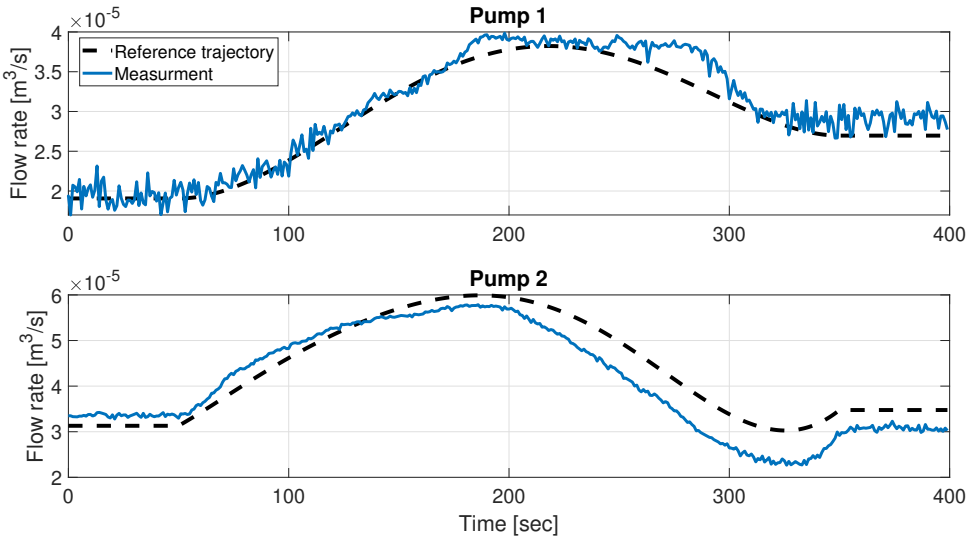

FIGURE 4.36: *Reference trajectories vs. measurements of the flow rate of each Pump P_i*

4.5.2 Derivative estimation and threshold determination

The components of the considered flat output of the system must be derived in order to calculate the redundant values y^z and u^z, see (3.67) and (3.68). However, these derivatives may not exist because of the presence of noise on system sensors and actuators. So these derivatives have to be estimated. In DTS200 three-tank model, the noise on the measurements are the water bubbles due to the water falling

from the top of tanks T_1 and T_2 through pumps P_1 and P_2. In our experiments, we use, as in simulations, a *Butterworth Low-pass Filter* of order $N = 4$ and cutoff frequency $f_c = 0.35Hz$ to filter noise, together with a *Discrete Filtered Derivative* represented by the following transfer function (Group et al., 1992):

$$\left(\frac{K}{T}\right)\frac{z-1}{z+Ts/T-1} \tag{4.54}$$

where $K = 1$ is the gain, $T = 20s$ is the time constant and $Ts = 1s$ is the sample time.

Moreover, a threshold is fixed for each residue in order to avoid false alarm. For the purpose of setting thresholds, several nominal experiments were run, *i.e.* without introducing any fault on system sensors and actuators. Each time the initial and final conditions are modified. The maximum and minimum values of the residues are extracted in each experiment, and the amplitude of the threshold is fixed by choosing the worst case among all the calculated residues. A safety margin of 5% is added to avoid false alarms. The threshold values of each residue are given in Table 4.7.

	Max	Min
$R^{Z_1}_{S_2}\ [m]$	0.0273	-0.0141
$R^{Z_1}_{A_1}\ [m^3/s]$	6.802×10^{-6}	-9.1133×10^{-6}
$R^{Z_1}_{A_2}\ [m^3/s]$	2.6309×10^{-5}	-3.1967×10^{-5}
$R^{Z_2}_{S_1}\ [m]$	0.0256	-0.0273
$R^{Z_2}_{A_1}\ [m^3/s]$	3.7157×10^{-5}	-2.5906×10^{-5}
$R^{Z_2}_{A_2}\ [m^3/s]$	1.6906×10^{-5}	-1.8138×10^{-5}

TABLE 4.7: *Experimental values of the maximum and minimum threshold for each residue*

4.5.3 Experimental FDI results

The same type of faults as those of the simulations were applied to the experiments. Recall that we have only one fault at a time.

Sensor faults:

Figure 4.37 shows that a fault that affects the sensor S_1, affects only the residues that depends on the measurements y_1^s of S_1, which confirms the fault alarm signature Σ_1 of the augmented signature matrix $\widetilde{\mathbf{S}}$ (equation (4.45)):

$$\Sigma_1 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}^T . \tag{4.55}$$

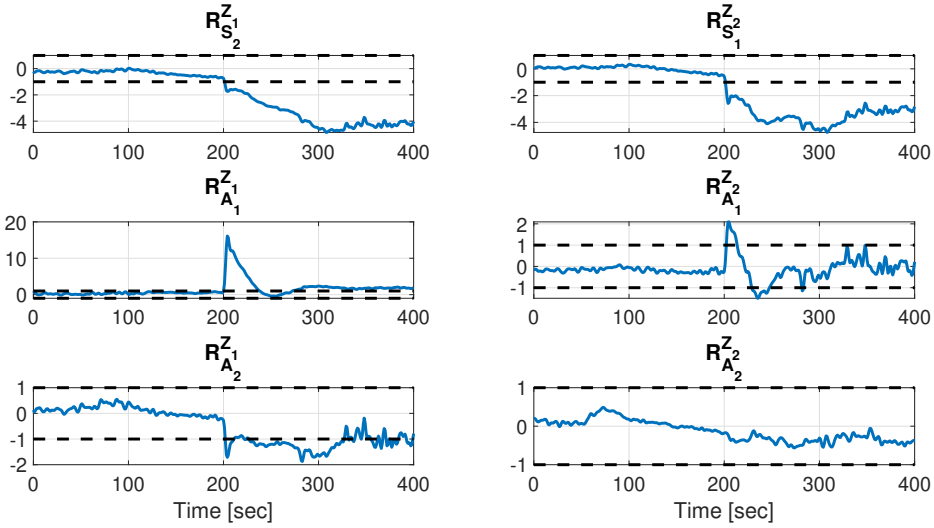

FIGURE 4.37: *Experimental results: multiplicative fault on sensor* S_1 *at time* $t = 200s$

Likewise, a fault that affects the sensor S_2, affects only the residues that depend on the measurement y_2^s, see Figure 4.38. Hence the signature Σ_2 of the augmented signature matrix $\widetilde{\mathbf{S}}$:

$$\Sigma_2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}^T . \tag{4.56}$$

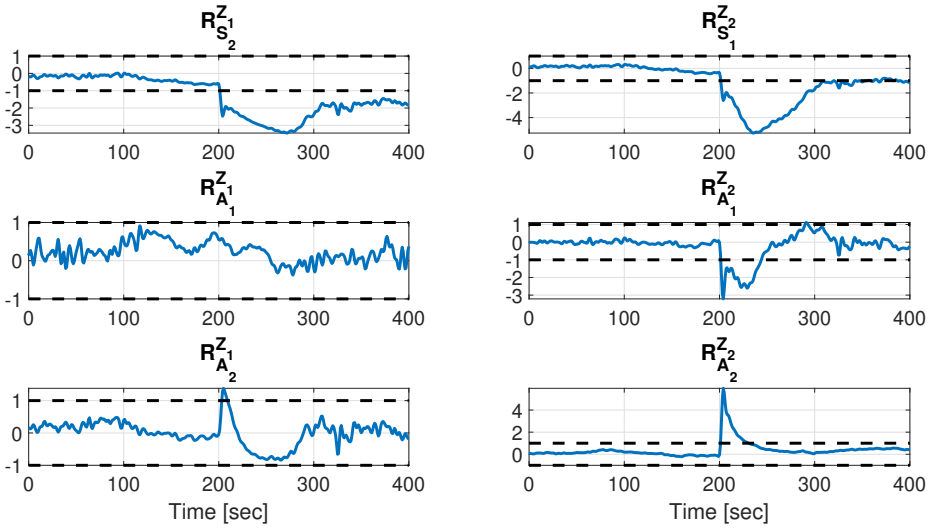

FIGURE 4.38: *Experimental results: multiplicative fault on sensor* S_2 *at time* $t = 200s$

A fault on sensor S_3 must affect all the residues, since the measurement y_3^s and its successive time derivatives are parts of these residues. This result has also been obtained by simulations (see Figure 4.31). However, in our experiments, even if residues $R^{Z_1}_{A_1}$ and $R^{Z_2}_{A_2}$ change their amplitudes at time $t = 200s$, it is not sufficient to exceed their thresholds, see Figure 4.39. The fault alarm signature in this case is

$$\Sigma_3 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}^T . \tag{4.57}$$

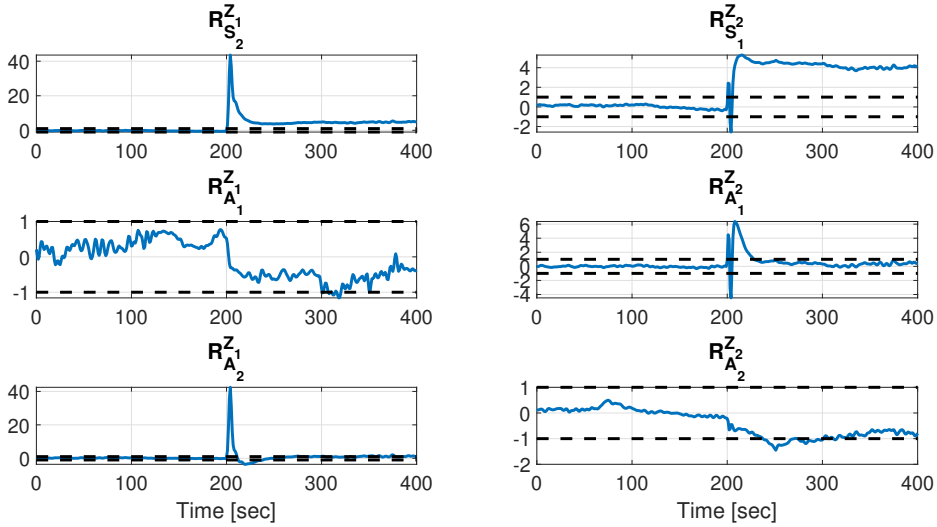

FIGURE 4.39: *Experimental results: multiplicative fault on sensor* S_3 *at time* $t = 200s$

Uncertainties on the system force us to choose higher thresholds (see Table 4.5 and Table 4.7) and therefore, these thresholds may no longer be exceeded in the event of a fault. Then, it is clear that this unexpected behaviour in Figure 4.39 is due to the presence of uncertainties on the experimental platform. Moreover, back to Figure 4.36, we can see that inputs u_1 and u_2 do not follow their reference trajectories even in the case of absence of faults. Which also explains the presence of uncertainties on the system. Therefore, this flatness-based FDI method is sensitive to uncertainties. And perhaps a small change in the parameters can cause false alarms, or missed detection of faults, as in our case.

Actuator faults:

Theoretically, faults on pumps P_1 and P_2 affect only the residues that depend on the measurements u_1 and u_2, respectively. Therefore, fault on pump P_1 affects residues $R^{Z_1}_{A_1}$ and $R^{Z_2}_{A_1}$. This result is confirmed experimentally, see Figure 4.40. The fault alarm signature in this case is

$$\Sigma_4 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}^T . \tag{4.58}$$

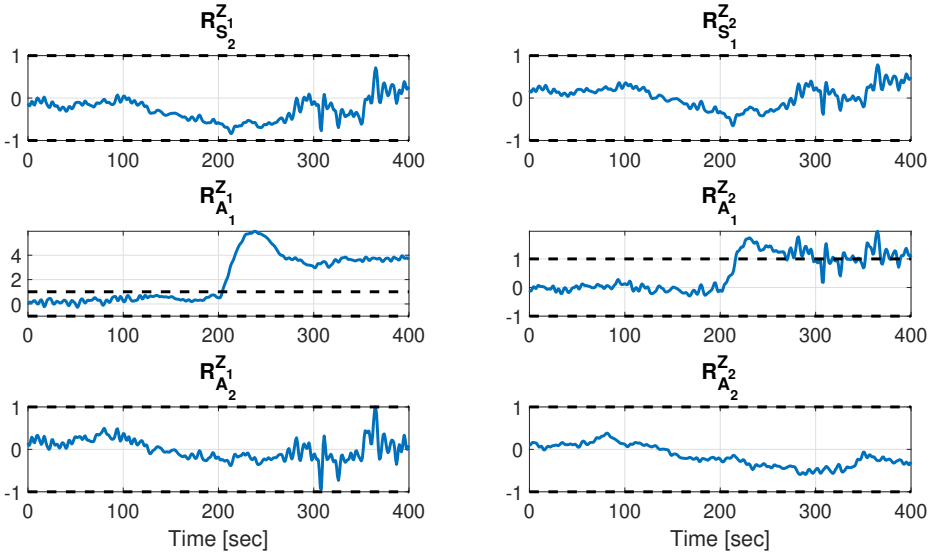

FIGURE 4.40: *Experimental results: multiplicative fault on pump P_1 at time $t = 200s$*

Likewise, a fault on pump P_2 affects only residues $R^{Z_1}_{A_2}$ and $R^{Z_2}_{A_2}$, see Figure 4.41. The fault alarm signature is then given by:

$$\Sigma_5 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}^T. \tag{4.59}$$

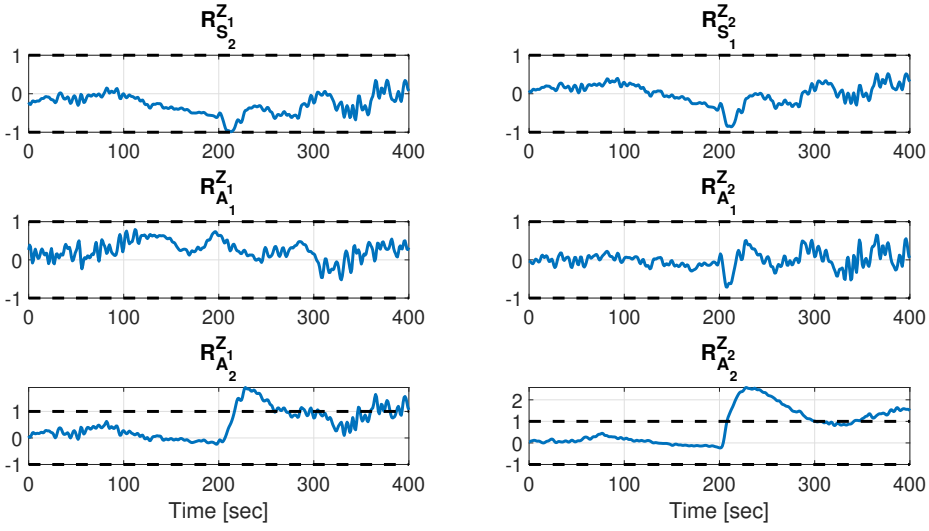

FIGURE 4.41: *Experimental results: multiplicative fault on pump P_1 at time $t = 200s$*

4.6 Conclusion

In this chapter, an application of the flat output characterization, proposed in Chapter 3 for the aim of FDI, was performed on the three-tank system. In particular, we showed that by using a single flat output, total fault isolability was not achieved. Then, we used another flat output vector and we verified that these two flat outputs are independent. Therefore, by using them together, we obtained a total fault isolability.

The effectiveness of this characterization has also been demonstrated by simulations. Two different cases have been taken into account: the open-loop case and the closed-loop case, and we have shown how the addition of a controller can affect the generation of residual signals and then the isolability of faults.

In addition to these simulations, experiments on the real three-tank system have been carried out. These experiments performed on the experimental test-bench have shown that this method can be sensitive to uncertainties and hence requires further development in the future.

Part II

Diagnostic of Fractional-Order Linear Flat Systems

Chapter 5

Unimodular Completion Algorithm for Fractional Linear Systems

Abstract

The differential flatness has proven its efficiency in the field of automatic control systems over the last two decades. Methods for trajectory planning, trajectory tracking, system diagnostic and, recently, FTC and FDI have been developed using this property. The flatness was first introduced for the class of nonlinear integer-order systems as a generalization of the controllability property of linear systems, and then it was extended to cover the class of linear fractional-order systems. A dynamic system is said to be flat if, and only if, all the system variables can be expressed as function of a variable called flat output and its successive time derivatives. Thus, the interest of the flatness lies in the calculation of the expression of the flat output. In this chapter, we recall the differential flatness of fractional linear systems and we introduce a new algorithm of computation of the *fractional flat output*, based on the extension of the unimodular completion algorithm, presented in Chapter 3.

5.1 Introduction

Controllability property (2.7) and Brunovskỳ's canonical form (2.8), related to linear systems

$$\dot{x} = Ax + Bu \tag{5.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector and $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ are real matrices, represent a first approach to solve the problem of path planning and path tracking. Path planning consists in building offline trajectories, called reference trajectories, with associated inputs, based on knowledge of the model and without disturbances. These trajectories link an initial point to a final point in open loop. However, the path tracking consists on the design of a control law allowing to follow the reference trajectory.

In the case of nonlinear systems

$$\dot{x} = f(x, u) \tag{5.2}$$

these two aspects, "path planning" and "path tracking", are simple to solve if the nonlinear system verifies the flatness property or, in other words, if the system is flat. Then, the flatness property is a generalization of the controllability property of linear systems to nonlinear systems. The method of generation of reference trajectories using the flatness property is detailed in Section 2.5.

For linear systems, it turns out that these systems are flat if, and only if, they are controllable (Fliess et al., 1999). In this case, there exists a variable called "linearizing output" or "flat output", derived from Brunovskỳ's canonical form, such that all the system variables are written as a function of this flat output and its successive time derivatives.

In Victor (2010), an extension of the controllability to non-integer or fractional linear systems is established based on the module theory. Using the module theory, the controllability is independent of any particular representation of the system. Moreover, the equivalence between controllability and flatness for fractional linear systems has been shown, and the fractional flat output is the variable resulting from the Brunovský's canonical form.

The main challenge of the flatness concept is the computation of flat outputs. A method of computation of fractional flat outputs, based on the Smith decomposition of $\mathbf{D}_a^\gamma$-polynomial matrix has been developed by Victor et al. (2015) and recalled in section 5.2.4. In this chapter, we develop a new method of computation of fractional flat outputs, based on the extension of the unimodular completion algorithm for integer-order systems, developed by Fritzsche et al. (2016a) and recalled in Section 3.2.2.

5.2 Fractional linear flatness

In order to recall the property of flatness for fractional linear systems, we first mention some notions of fractional calculus.

5.2.1 Fractional calculus

The fractional derivative appeared in the $19th$ century as a generalization of the traditionally used derivative, see Euler (1738), Fourier (1822), Laplace (Stigler, 2005), Liouville (1832) and Riemann, Dedekind, and Weber (1892). For more recent references, see Miller and Ross (1993), Samko, Kilbas, Marichev, et al. (1993) and Dugowson (1994). However, it has been regarded solely as a theoretical notion until the discovery of physical systems that can be modeled by fractional differential equations (Trigeassou and Maamri, 2019), such as thermal systems (Battaglia et al., 2000), nuclear magnetic resonance systems (Magin et al., 2008) and viscoelastic systems (Moreau, Ramus-Serment, and Oustaloup, 2002). This notion of fractional calculation is also useful in robust control, such as in the CRONE control (Oustaloup, 1995).

5.2.1.1 Fractional integral

Let $n \in \mathbb{N}^*$ be a non zero integer and $f(t) \in \mathcal{C}^\infty([a, +\infty[)$ the set of infinitely continuously differentiable functions. The integration of order n of the function

$f(t)$ is defined, according to Cauchy's formula, by:

$$\mathbf{I}_a^n f(t) = \frac{1}{(n-1)!}\int_a^t \frac{f(\tau)}{(t-\tau)^{1-n}}d\tau. \tag{5.3}$$

The generalization of the Cauchy formula to the fractional integration of order $\gamma \in \mathbb{R}_+$ has been defined by Riemann and Liouville (Miller and Ross, 1993):

$$\mathbf{I}_a^\gamma f(t) = \frac{1}{\Gamma(\gamma)}\int_a^t \frac{f(\tau)}{(t-\tau)^{1-\gamma}}d\tau, \tag{5.4}$$

where

$$\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt, \quad \forall x \in \mathbb{R}^* \setminus \mathbb{N}^- \tag{5.5}$$

is the Euler's function or the generalized factorial. Note that for all $n \in \mathbb{N}$

$$\Gamma(n+1) = n!, \tag{5.6}$$

and for all $\gamma \in \mathbb{R}_+$

$$\gamma\Gamma(\gamma) = \Gamma(\gamma+1). \tag{5.7}$$

This integral can be interpreted geometrically as the area of the surface defined by the function $f(t)$ weighted by the function $\mathcal{O}_\gamma = \frac{(t-\tau)^{\gamma-1}}{\Gamma(\gamma)}$. For the integer order $\gamma = 1$, $\mathcal{O}_\gamma = 1$ and the integral $\mathbf{I}_a^\gamma$ corresponds to the area of the surface between f and the abscissa axis on τ over $[0, t]$. For a non-integer γ, the function $\mathcal{O}_\gamma$ weighs differently the function f: indeed, the value of the integral of order $\gamma \in]0, 1[$ at a point t is more influenced by points in its neighborhood than by points further away (see Figure 5.1). As a result, the weighting function $\mathcal{O}_\gamma$ is also referred to as the forgetting factor (Oustaloup, 1995). This consideration of the past demonstrates the natural ability of the fractional integration operator to describe long memory phenomena such as diffusion phenomena.

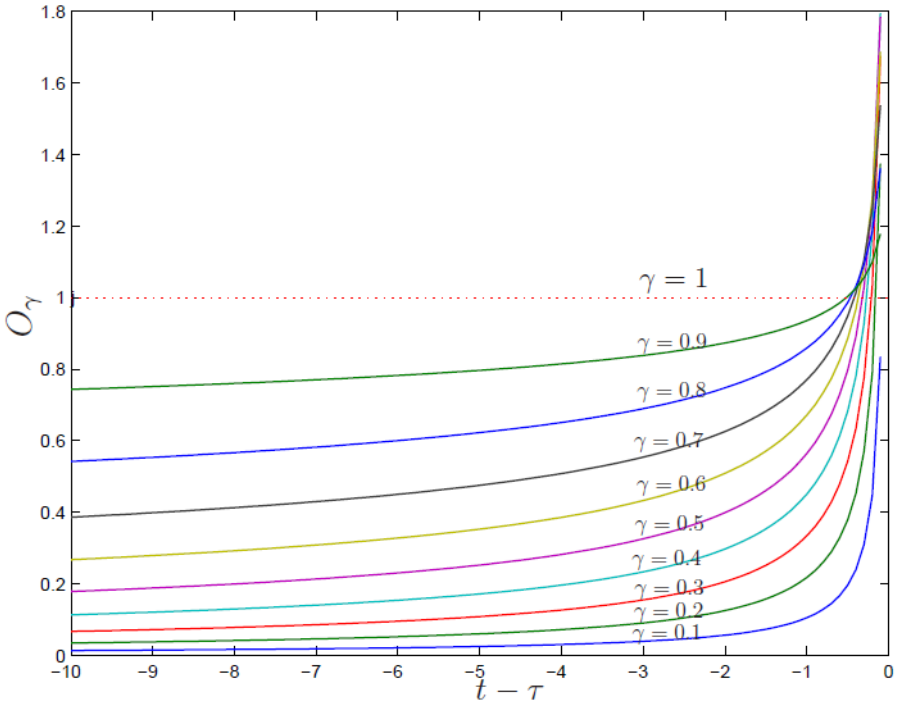

FIGURE 5.1: *Forgetting factor $\mathcal{O}_\gamma$ for $\gamma \in]0,1[$, Source: Victor (2010)*

5.2.1.2 Fractional derivative

Let $\gamma \in \mathbb{R}_+$ be a positive real number, $n = \min\{k \in \mathbb{N} \mid k > \gamma\}$ the smallest integer greater than γ and $\nu = n - \gamma \in [0,1[$.

Definition 5.2.1 (Miller and Ross, 1993) *The fractional derivative of order $\gamma = n - \nu$ of a function $f \in \mathcal{C}^\infty([a,+\infty[)$ at time t, denoted by $\mathbf{D}_a^\gamma f(t)$, is defined by the n^{th} order derivative of the fractional integral of order ν:*

$$\mathbf{D}_a^\gamma f(t) = \mathbf{D}^n\left(\mathbf{I}_a^\nu f(t)\right) \triangleq \left(\frac{d}{dt}\right)^n \left(\frac{1}{\Gamma(\nu)} \int_a^t \frac{f(\tau)}{(t-\tau)^{1-\nu}} d\tau\right). \tag{5.8}$$

For particular values of the reference a, the fractional derivative (5.8) becomes:

- Liouville derivative if $a = -\infty$;
- Riemann derivative if $a \neq 0$;
- Riemann-Liouville derivative if $a = 0$.

If $\gamma = n \in \mathbb{N}$, the fractional derivative coincides with the ordinary derivative:

$$\mathbf{D}_a^{\gamma} f(t) = \mathbf{D}_a^n f(t).$$

If $\gamma < 0$, the fractional derivative is in fact the fractional integral:

$$\mathbf{D}_a^{\gamma} f(t) = \mathbf{I}_a^{-\gamma} f(t).$$

Properties of differentiability, integrability and commutativity of the fractional operator can be found in Podlubny (1999). Moreover, according to Miller and Ross (1993), the operator $\mathbf{D}_a^{\gamma}$ is a linear operator:

Proposition 5.2.1 *Let f and $g \in \mathcal{C}^{\infty}([a, +\infty[)$ and α and $\beta \in \mathbb{R}$, we have:*

$$\mathbf{D}_a^{\gamma}(\alpha f(t) + \beta g(t)) = \alpha \mathbf{D}_a^{\gamma} f(t) + \beta \mathbf{D}_a^{\gamma} g(t). \tag{5.9}$$

The Laplace transform of the fractional derivative of a function f at time t, in the sense of Riemann-Liouville is defined by Miller and Ross (1993):

$$\mathscr{L}(\mathbf{D}_a^{\gamma} f(t))(s) = s^{\gamma} F(s) - \sum_{k=0}^{n} s^{n-k} \mathbf{D}_a^k (\mathbf{I}_a^{n+1-\gamma} f(t))\Big|_{t=a} \tag{5.10}$$

where $F(s)$ is the Laplace transform of $f(t)$. In system theory, the signal space is defined as the space of causal functions $\mathfrak{H}_a$ given by:

$$\mathfrak{H}_a \triangleq \{f : \mathbb{R} \mapsto \mathbb{R} \mid f \in \mathcal{C}^{\infty}([a, +\infty[), f(t) = 0, \forall t \leq a\}. \tag{5.11}$$

In this case, the expression (5.10) of the Laplace transform is reduced to:

$$\mathscr{L}(\mathbf{D}_a^{\gamma} f(t))(s) = s^{\gamma} F(s). \tag{5.12}$$

The operator $\mathbf{D}_a^{\gamma}$ is an endomorphism from $\mathfrak{H}_a$ to $\mathfrak{H}_a$ (see Podlubny, 1999).

5.2.1.3 Representation of fractional linear systems

There exist two levels of generalization of linear systems of the form (5.1) to fractional linear system. In the first one, the orders of derivation of all the elementary differential equations are the same, *i.e.* $x^{(\nu)} = (x_1^{(\nu)}, \ldots, x_n^{(\nu)})$. The pseudo-state representation is then given by:

$$x^{(\nu)} = Ax + Bu \tag{5.13}$$

where $x \in (\mathfrak{H}_a)^n$ represents the n-dimensional pseudo-state vector[1].

In the second one, each elementary differential equation has a different order of derivation. Consider the vector $\underline{\nu} = (\nu_1, \ldots, \nu_n)$ of dimension n, the pseudo-state representation in this case is of the form:

$$x^{(\underline{\nu})} = Ax + Bu \tag{5.14}$$

where the derivation of the component x_i is of order ν_i, *i.e.* $x^{(\underline{\nu})} = (x_1^{(\nu_1)}, \ldots, x_n^{(\nu_n)})$.

For these systems the problem of the initial conditions is more complicated than for integer-order systems. For details on the initialization problem of fractional order systems see Trigeassou and Maamri (2019). In the sequel, we suppose that the state, input and output of the system are in $\mathfrak{H}_a$, *i.e.* the initial conditions are zero.

5.2.1.4 Controllability of fractional linear system

In the following, we consider the fractional linear system of the first level of generalization (5.13). The controllability property of these systems is given by the following theorem:

Theorem 5.2.1 (Matignon and Novel, 1996) *The system* (5.13) *is controllable if, and only if*

$$\text{rank}\,(B, AB, \ldots, A^{n-1}B) = n. \tag{5.15}$$

[1]This notation is specified for the class of fractional system and it refers to Oustaloup (1995).

In this case, there exists a change of coordinates, or a static state feedback, that transforms the system (5.13) into the form:

$$\begin{cases} z_1^{(\nu_1)} = v_1 \\ z_2^{(\nu_2)} = v_2 \\ \quad \vdots \\ z_m^{(\nu_m)} = v_m \end{cases} \tag{5.16}$$

where v_i are the new input variables and ν_i are the controllability indices which are non-integer. The set $z = (z_1, \ldots, z_m)$ is called fractional flat output. Then, according to Victor (2010), the system (5.13) is flat if, and only if, it is controllable.

The property of observability of such systems can be found in Fliess and Hotzel (1997) and of stability can be found in Victor (2010).

5.2.2 Fractional linear flat system: a polynomial approach

In order to extend the definition of flatness from linear systems to fractional linear systems, Victor (2010, Chapter 1) has extended the $\frac{d}{dt}$-polynomials and $\frac{d}{dt}$-polynomial matrices to $\mathbf{D}_a^\gamma$-polynomials and $\mathbf{D}_a^\gamma$-polynomial matrices as follows: let $\mathbb{R}[\mathbf{D}_a^\gamma]$ be the set of $\mathbf{D}_a^\gamma$-polynomials with real coefficients of the form

$$\sum_{k=0}^{K} c_k \mathbf{D}_a^{k\gamma} = c_0 + c_1 \mathbf{D}_a^\gamma + \ldots + c_K \mathbf{D}_a^{K\gamma}. \tag{5.17}$$

This set, endowed with the usual addition and multiplication of polynomials $(\mathbb{R}[\mathbf{D}_a^\gamma], +, \times)$, is a commutative principal ideal domain. Let p and $q \in \mathbb{N}$, we denote by $\mathbb{R}[\mathbf{D}_a^\gamma]^{p\times q}$ the set of matrices of size $p \times q$, whose entries are $\mathbf{D}_a^\gamma$-polynomials. An invertible square matrix of $\mathbb{R}[\mathbf{D}_a^\gamma]^{p\times p}$ whose inverse is also in $\mathbb{R}[\mathbf{D}_a^\gamma]^{p\times p}$ is called unimodular matrix. The set of unimodular matrices is denoted by $GL_p(\mathbb{R}[\mathbf{D}_a^\gamma])$.

The Smith decomposition has also been proved in Victor et al. (2015) for $\mathbf{D}_a^\gamma$-polynomial matrices:

Theorem 5.2.2 (Smith decomposition) *Let $M \in \mathbb{R}[\mathbf{D}_a^\gamma]^{p\times q}$ with $p \leq q$ (resp. $p \geq q$), then there exist two matrices $S \in GL_p(\mathbb{R}[\mathbf{D}_a^\gamma])$ and $T \in GL_q(\mathbb{R}[\mathbf{D}_a^\gamma])$ and a matrix $\Delta = \text{diag}\,\{\delta_1, \ldots, \delta_\sigma, 0, \ldots, 0\} \in \mathbb{R}[\mathbf{D}_a^\gamma]^{p\times p}$ (resp. $\mathbb{R}[\mathbf{D}_a^\gamma]^{q\times q}$) such that:*

$$SMT = \begin{pmatrix} \Delta & 0_{p\times(q-p)} \end{pmatrix} \quad \left(resp.\ \ SMT = \begin{pmatrix} \Delta \\ 0_{(p-q)\times q} \end{pmatrix}\right). \tag{5.18}$$

In Δ, $\sigma = \text{rank}\,(M) \leq \min(p,q)$ and δ_i, for $i = 1, \ldots, \sigma$, is a $\mathbf{D}_a^\gamma$-polynomial such that δ_i divides δ_j for all $i \leq j \leq \sigma$.

$\mathbf{D}_a^\gamma$-polynomial matrices admit also the following property (Antritter et al., 2014):

Theorem 5.2.3 (Hyper-regularity) *A matrix $M \in \mathbb{R}[\mathbf{D}_a^\gamma]^{p\times q}$ with $p \leq q$ (resp. $p \geq q$) is said to be hyper-regular if, and only if, there exists a matrix $U \in GL_p(\mathbb{R}[\mathbf{D}_a^\gamma])$ (resp. $V \in GL_q(\mathbb{R}[\mathbf{D}_a^\gamma])$) such that:*

$$MU = \begin{pmatrix} I_p & 0_{p\times(q-p)} \end{pmatrix} \quad \left(resp.\ \ VM = \begin{pmatrix} I_q \\ 0_{(p-q)\times q} \end{pmatrix}\right). \tag{5.19}$$

The system (5.13) can be written into the form:

$$\mathcal{A}x = \mathcal{B}u \tag{5.20}$$

where $\mathcal{A} \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times n}$ and $\mathcal{B} \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times m}$ are $\mathbf{D}_a^\gamma$-polynomial matrices. The matrix $\mathcal{B}$ is supposed to be of rank m and $m \leq n$. In turn, the system (5.20) can be transformed into the form:

$$F \begin{pmatrix} x \\ u \end{pmatrix} = 0 \tag{5.21}$$

where $F \triangleq \begin{pmatrix} \mathcal{A} & -\mathcal{B} \end{pmatrix} \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times(n+m)}$ is assumed to be of full row rank.

Inspired by the work of Antritter et al. (2014) on the flatness of linear systems, the definition of fractional linear flatness is then introduced by Victor et al. (2015) as follows:

Definition 5.2.2 *The system* (5.21) *is said to be fractionally flat if, and only if, there exist two matrices* $P \in \mathbb{R}[\mathbf{D}_a^\gamma]^{m\times(n+m)}$ *and* $Q \in \mathbb{R}[\mathbf{D}_a^\gamma]^{(n+m)\times m}$ *and a variable* $z \in (\mathfrak{H}_a)^m$ *such that:*

1. $PQ = I_m$;
2. *For all* $(x,u)^T$ *satisfying* (5.21), *we have* $z = P\begin{pmatrix} x \\ u \end{pmatrix}$ *and conversely* $\begin{pmatrix} x \\ u \end{pmatrix} = Qz$.

The variable z *is called fractional flat output and the matrices* P *and* Q *are called fractional defining matrices.*

The main property of fractional linear flatness is given by the following theorem (Victor et al., 2015):

Theorem 5.2.4 *The system* (5.21) *is fractionally flat if, and only if, the matrix* F *given by* (5.21) *is hyper-regular over* $\mathbb{R}[\mathbf{D}_a^\gamma]$.

In some cases, the system (5.21) admits an implicit form as follows:

Proposition 5.2.2 (Implicit Form) *If* $B \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times m}$ *is hyper-regular, i.e. if there exists* $M \in GL_n(\mathbb{R}[\mathbf{D}_a^\gamma])$ *such that* $MB = \begin{pmatrix} I_m \\ 0_{(n-m)\times m} \end{pmatrix}$, *then there exist two matrices* $\widetilde{F} \in \mathbb{R}[\mathbf{D}_a^\gamma]^{(n-m)\times n}$ *and* $R \in \mathbb{R}[\mathbf{D}_a^\gamma]^{m\times n}$ *such that the system* (5.21) *is equivalent*

$$\begin{cases} Rx = u \\ \widetilde{F}x = 0 \end{cases}. \tag{5.22}$$

The proof of this proposition is the same as in Victor et al. (2015). Equation $\widetilde{F}x = 0$ is called the fractional implicit system.

Remark 5.2.1 *In practice, the fractional flat output may depends only on the state variable x. More precisely, the defining matrix P of the Definition 5.2.2 may be of the form $P = [P_1 \;\; 0_m]$ with $P_1 \in \mathbb{R}[\mathbf{D}_a^\gamma]^{m\times n}$ and the fractional flat output is then given by $z = P_1 x$. In this case we say that the system is fractionally (-1)-flat (Victor et al., 2015).*

Therefore, the property of flatness can be transferred to the class of fractional implicit linear systems, according to the following theorem:

Theorem 5.2.5 *Let the matrix B of the system* (5.21) *be hyper-regular, then the system* (5.21) *is fractionally (-1)-flat if, and only if, the matrix $\widetilde{F}$ of the implicit form is hyper-regular over $\mathbb{R}[\mathbf{D}_a^\gamma]$.*

5.2.3 Fractional nonlinear flat systems

The flatness property has been, in fact, introduced to deal with nonlinear systems, since for linear systems this property is equivalent to the controllability property. Works in Victor (2010) have shown that the extension of the flatness property to the class of fractional nonlinear systems is still not possible due to the absence of some necessary mathematical tools.

The fractional nonlinear systems are of the form:

$$x^{(\nu)} = f(x, u) \tag{5.23}$$

where $x = (x_1, \ldots, x_n)$ is the pseudo-state vector, $u = (u_1, \ldots, u_m)$ is the input vector and ν the fractional order derivative. In the literature, there exists two approaches for the nonlinear flatness: differential algebra (Kolchin, 1973; Ritt, 1950) and differential geometry of jets of infinite order (Krasilchchik, Vinogradov, and Lychagin, 1996; Bocharov, Shchik, and Vinogradov, 1999). However, the differential geometry provides necessary and sufficient conditions for the nonlinear flatness (Levine, 2009). The idea in Victor (2010) was to extend this approach to the class of fractional nonlinear systems (5.23). Studies were done via two ways: using the standard differential operator d (Levine, 2009), and using the fractional differential operator d_ν (Cottrill-Shepherd and Naber, 2001). In both ways, the extension to the fractional order is not consistent with the integer case. Thus until now, the

property of flatness for fractional nonlinear systems is still an open problem and mathematical tools must be developed first.

5.2.4 Computation of the fractional flat output

The flatness is characterized by the existence of a variable called flat output, formed by the elements of the state and the input vectors of the system, and such that all the variables of the system can be expressed as a function of this flat output and its successive derivatives. This characterization facilitates the problem of trajectory planning and trajectory tracking. Then, the main task would be to compute the expressions of these flat outputs.

An algorithm of computation of the fractional flat output has been developed in Victor et al. (2015) and based on the Smith decomposition of $\mathbf{D}_a^\gamma$-polynomial matrices. We recall that for the class of fractional linear systems, the pseudo-state vector $x \in \mathfrak{H}_a^n$ and the input vector $u \in \mathfrak{H}_a^m$. In order to compute defining matrices P and Q and a fractional flat output z of the Definition 5.2.2, we suppose that F is hyper-regular, then, by the Smith decomposition, we compute $W \in GL_{n+m}(\mathbb{R}[\mathbf{D}_a^\gamma])$ such that:

$$FW = \begin{pmatrix} I_n & 0_{n\times m} \end{pmatrix}. \tag{5.24}$$

The defining matrices Q and P are then given by:

$$Q = W \begin{pmatrix} 0_{n\times m} \\ I_m \end{pmatrix} \quad \text{and} \quad P = \begin{pmatrix} 0_{m\times n} & I_m \end{pmatrix} W^{-1} \tag{5.25}$$

respectively. Therefore, a fractional flat output vector is given by $z = P \begin{pmatrix} x \\ u \end{pmatrix}$ and conversely we have $\begin{pmatrix} x \\ u \end{pmatrix} = Q\,z$.

In the next section, a new method of computation of fractional flat outputs is developed, based on the notion of unimodular completion of $\mathbf{D}_a^\gamma$-polynomial matrix. In this method, decomposition of coefficient matrices are used instead of $\mathbf{D}_a^\gamma$-polynomial matrices. Moreover, this method provides, in some particular cases, a direct representation of fractional flat outputs, *i.e.* without the need to make calculations, called fractionally direct flat representation.

5.3 Unimodular completion algorithm

The unimodular completion algorithm for the computation of flat outputs has been recalled for the class of nonlinear systems in Section 3.2.2. In this section, an extension of this algorithm to the class of fractionally linear flat systems is developed using some properties of the fractional calculus.

5.3.1 Preliminary definitions

Definition 5.3.1 *Given a hyper-regular matrix $M \in \mathbb{R}[\mathbf{D}_a^\gamma]^{p\times q}$ with $p \leq q$, we say that $N \in \mathbb{R}[\mathbf{D}_a^\gamma]^{(q-p)\times q}$ is a unimodular completion of M if, and only if,*

$$\begin{pmatrix} M \\ N \end{pmatrix} \in GL_q(\mathbb{R}[\mathbf{D}_a^\gamma]).$$

Proposition 5.3.1 *Let F defined by* (5.21) *be hyper-regular. Then, the vector z is a fractional flat output of* (5.21) *if, and only if, the matrix $P \in \mathbb{R}[\mathbf{D}_a^\gamma]^{m\times(n+m)}$ such that*

$$z = P \begin{pmatrix} x \\ u \end{pmatrix} \tag{5.26}$$

is a unimodular completion of F.

Proof 1 *The matrix $F \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times(n+m)}$ is hyper-regular, then, by the Smith decomposition of F, there exists a matrix $W \in GL_{n+m}(\mathbb{R}[\mathbf{D}_a^\gamma])$ such that:*

$$FW = \begin{pmatrix} I_n & 0_{n\times m} \end{pmatrix} \tag{5.27}$$

which implies that $F = \begin{pmatrix} I_n & 0_{n\times m} \end{pmatrix} W^{-1}$, and F constitutes the first n rows of W^{-1}. In addition, the defining matrix P given by $P = \begin{pmatrix} 0_{m\times n} & I_m \end{pmatrix} W^{-1}$ constitutes the last m rows of W^{-1}, then we get:

$$W^{-1} = \begin{pmatrix} F \\ \\ P \end{pmatrix} \quad \in GL_{n+m}(\mathbb{R}[\mathbf{D}_a^\gamma])$$

and P is a unimodular completion of F. □

In the case where the matrix B is hyper-regular, *i.e.* the system (5.21) is fractionally (-1)-flat, Proposition 5.3.1 becomes:

Proposition 5.3.2 *The vector z is a fractional flat output of the implicit system* (5.22) *if, and only if, the matrix $P \in \mathbb{R}[\mathbf{D}_a^\gamma]^{m\times(n+m)}$ such that*

$$z = P \begin{pmatrix} x \\ \\ u \end{pmatrix} \tag{5.28}$$

is a unimodular completion of $\widetilde{F}$.

The proof of this proposition is the same as for Proposition 5.3.1.

5.3.2 Computation procedure

Notations used in the following are the same adapted in Fritzsche et al. (2016a). The algorithm is iterative and consists of three steps: Reduction, Zero-space Decomposition and Elimination. The starting point is the system (5.21) which can be

decomposed into the form

$$\left(F_{0,[0]} + F_{1,[0]}\ \mathbf{D}_a^{\gamma}\right) v_{[0]} = 0 \tag{5.29}$$

where $F_{0,[0]}$ and $F_{1,[0]}$ in $\mathbb{R}^{n\times(n+m)}$ are two coefficient matrices and $v_{[0]} = (x,u)^T$. The index in brackets indicates the iteration number.

5.3.2.1 Reduction

Starting from

$$\left(F_{0,[i]} + F_{1,[i]}\ \mathbf{D}_a^{\gamma}\right) v_{[i]} = 0 \tag{5.30}$$

for some iteration i, we consider the change of coordinates

$$v_{[i]} = F_{1,[i]}^{\dagger R}\, v_{[i+1]} + F_{1,[i]}^{\perp R}\, w_{[i+1]} \tag{5.31}$$

where $F_{1,[i]}^{\dagger R}$ is the right pseudo-inverse:

$$F_{1,[i]} F_{1,[i]}^{\dagger R} = I_n \tag{5.32}$$

and $F_{1,[i]}^{\perp R}$ is the right-orthonormal:

$$F_{1,[i]} F_{1,[i]}^{\perp R} = 0. \tag{5.33}$$

By injecting equation (5.31) in (5.30) and using the property (5.9), we get

$$v_{[i+1]}^{(\gamma)} + \mathfrak{A}_{[i]} v_{[i+1]} + \mathfrak{B}_{[i]} w_{[i+1]} = 0 \tag{5.34}$$

with

$$\mathfrak{A}_{[i]} = F_{0,[i]} F_{1,[i]}^{\dagger R} \quad \text{and} \quad \mathfrak{B}_{[i]} = F_{0,[i]} F_{1,[i]}^{\perp R}. \tag{5.35}$$

The matrix $\mathfrak{B}_{[i]}$, being in $\mathbb{R}^{n_i\times m_i}$, two cases can be distinguished:

- If $rank\ (\mathfrak{B}_{[i]}) = r_i < m_i$ then a zero-space decomposition is needed to reduce the dimension.

- If $rank\ (\mathfrak{B}_{[i]}) = m_i$, *i.e.* $\mathfrak{B}_{[i]}$ is of full column rank, we move on to the Elimination step.

Remark 5.3.1 *If in* (5.35) *the matrix* $\mathfrak{B}_{[i]} \equiv 0$ *then the system* (5.34) *becomes:*

$$v^{(\gamma)}_{[i+1]} + \mathfrak{A}_{[i]} v_{[i+1]} = 0. \tag{5.36}$$

This system is no longer controllable and then the system is not fractionally flat (Franke and Röbenack, 2013).

5.3.2.2 Zero-space decomposition

As mentioned above, if $rank\ (\mathfrak{B}_{[i]}) = r_i < m_i$, then it is necessary to decompose the matrix $F^{\perp R}_{1,[i]}$ into the form

$$F^{\perp R}_{1,[i]} = \begin{pmatrix} \widetilde{F}^{\perp R}_{1,[i]} & Z_{[i]} \end{pmatrix} \tag{5.37}$$

such that

$$\mathfrak{B}_{[i]} = F_{0,[i]} \begin{pmatrix} \widetilde{F}^{\perp R}_{1,[i]} & Z_{[i]} \end{pmatrix} = \begin{pmatrix} \widetilde{\mathfrak{B}}_{[i]} & 0 \end{pmatrix} \tag{5.38}$$

with $rank\ (\widetilde{\mathfrak{B}}_{[i]}) = r_i$. For this purpose, the matrix $\mathfrak{B}_{[i]}$ is multiplied from the right by a transformation matrix $K \in GL_{m_i}(\mathbb{R}[\mathbf{D}^{\gamma}_a])$ consisting of two matrices K_1 and K_2 and such that:

$$\mathfrak{B}_{[i]} K = \mathfrak{B}_{[i]} \begin{pmatrix} K_1 & K_2 \end{pmatrix} = \begin{pmatrix} \widetilde{\mathfrak{B}}_{[i]} & 0 \end{pmatrix}. \tag{5.39}$$

On one side we have

$$\begin{aligned} \begin{pmatrix} \widetilde{\mathfrak{B}}_{[i]} & 0 \end{pmatrix} &= \mathfrak{B}_{[i]} \begin{pmatrix} K_1 & K_2 \end{pmatrix} \\ &= F_{0,[i]} F^{\perp R}_{1,[i]} \begin{pmatrix} K_1 & K_2 \end{pmatrix} \\ &= F_{0,[i]} \begin{pmatrix} F^{\perp R}_{1,[i]} K_1 & F^{\perp R}_{1,[i]} K_2 \end{pmatrix} \end{aligned} \tag{5.40}$$

and on the other side, we have

$$\begin{pmatrix} \widetilde{\mathfrak{B}}_{[i]} & 0 \end{pmatrix} = F_{0,[i]} \begin{pmatrix} \widetilde{F}^{\perp R}_{1,[i]} & Z_{[i]} \end{pmatrix} \tag{5.41}$$

then

$$\widetilde{F}_{1,[i]}^{\perp R} = F_{1,[i]}^{\perp R} K_1 \quad \text{and} \quad Z_{[i]} = F_{1,[i]}^{\perp R} K_2. \tag{5.42}$$

However, according to (5.39),

$$\mathfrak{B}_{[i]} \begin{pmatrix} K_1 & K_2 \end{pmatrix} = \begin{pmatrix} \widetilde{\mathfrak{B}}_{[i]} & 0 \end{pmatrix} \tag{5.43}$$

then, K_2 is the right orthonormal of $\mathfrak{B}_{[i]}$, *i.e.* $K_2 = \mathfrak{B}_{[i]}^{\perp R}$, and K_1 can then be used as a regular completion: $K_1 = \left(\left(\mathfrak{B}_{[i]}^{\perp R} \right)^{\perp L} \right)^T$. Hence, the expressions of $Z_{[i]}$ and $\widetilde{F}_{1,[i]}^{\perp R}$ are given by

$$Z_{[i]} := F_{1,[i]}^{\perp R} \mathfrak{B}_{[i]}^{\perp R} \quad \text{and} \quad \widetilde{F}_{1,[i]}^{\perp R} := F_{1,[i]}^{\perp R} \left(\left(\mathfrak{B}_{[i]}^{\perp R} \right)^{\perp L} \right)^T, \tag{5.44}$$

respectively. In this case, the change of coordinates (5.31) is replaced by

$$v_{[i]} = F_{1,[i]}^{\dagger R} \, v_{[i+1]} + \widetilde{F}_{1,[i]}^{\perp R} \, w_{[i+1]} + Z_{[i]} \, z_{[i+1]} \tag{5.45}$$

and equation (5.34) becomes:

$$v_{[i+1]}^{(\gamma)} + \mathfrak{A}_{[i]} v_{[i+1]} + \widetilde{\mathfrak{B}}_{[i]} w_{[i+1]} = 0. \tag{5.46}$$

5.3.2.3 Elimination

Returning to the Reduction step, if the matrix $\mathfrak{B}_{[i]}$ is not of full row rank, *i.e. rank* $(\mathfrak{B}_{[i]}) < n_i$, then the dimension of the system (5.30) must be reduced. For this purpose, the variable $w_{[i+1]}$ is eliminated from equation (5.34) by multiplying the latter by $\mathfrak{B}_{[i]}^{\perp L}$, which leads to:

$$\left(F_{0,[i+1]} + \mathbf{D}_a^{\gamma} \, F_{1,[i+1]} \right) v_{[i+1]} = 0, \tag{5.47}$$

with $F_{0,[i+1]} = \mathfrak{B}_{[i]}^{\perp L} \mathfrak{A}_{[i]}$ and $F_{1,[i+1]} = \mathfrak{B}_{[i]}^{\perp L}$. Here the system (5.47) is a reduced dimension of the system (5.30) and then the same procedure is repeated for the iteration $i+1$ on (5.47).

The calculations stop at iteration k when a full row rank of $\mathfrak{B}_{[k]}$ is reached.

Remark 5.3.2 *In the case where the zero-space decomposition is considered, the process of the Elimination step is applied to equation* (5.46) *by replacing* $\mathfrak{B}_{[i]}^{\perp L}$ *by* $\widetilde{\mathfrak{B}}_{[i]}^{\perp L}$.

5.3.2.4 Construction of the unimodular completion

In each iteration i, a relation between $v_{[i]}$ and $v_{[i+1]}$ can be deduced from (5.31) by left multiplying it by $F_{1,[i]}$:

$$v_{[i+1]} = F_{1,[i]} v_{[i]}. \tag{5.48}$$

Conversely, by multiplying (5.34) by $\mathfrak{B}_{[i]}^{\dagger L}$, $w_{[i+1]}$ can be expressed in function of $v_{[i+1]}$ as follows:

$$w_{[i+1]} = -\mathfrak{B}_{[i]}^{\dagger L} v_{[i+1]}^{(\gamma)} - \mathfrak{B}_{[i]}^{\dagger L} \mathfrak{A}_{[i]} v_{[i+1]}. \tag{5.49}$$

Then, by injecting (5.49) in (5.31) we get:

$$v_{[i]} = F_{1,[i]}^{\dagger R} v_{[i+1]} + F_{1,[i]}^{\perp R}(-\mathfrak{B}_{[i]}^{\dagger L} v_{[i+1]}^{(\gamma)} - \mathfrak{B}_{[i]}^{\dagger L} \mathfrak{B}_{[i]} v_{[i+1]}), \tag{5.50}$$

which leads to

$$\begin{aligned} v_{[i]} &= \big(F_{1,[i]}^{\dagger R} - F_{1,[i]}^{\perp R}(\mathfrak{B}_{[i]}^{\dagger L} \mathbf{D}_a^{(\gamma)} + \mathfrak{B}_{[i]}^{\dagger L} \mathfrak{A}_{[i]})\big) v_{[i+1]} \\ &= G_{[i]}(\mathbf{D}_a^{(\gamma)}) v_{[i+1]}. \end{aligned} \tag{5.51}$$

After a finite number $k+1$ of iterations, a relation between $v_{[k+1]}$ and $v_{[0]}$ is determined as follows:

$$v_{[k+1]} = F_{1,[k]} F_{1,[k-1]} \dots F_{1,[0]} v_{[0]} := P\, v_{[0]} \tag{5.52}$$

and

$$v_{[0]} = G_{[0]}(\mathbf{D}_a^{(\gamma)}) G_{[1]}(\mathbf{D}_a^{(\gamma)}) \dots G_{[k-1]}(\mathbf{D}_a^{(\gamma)}) G_{[k]}(\mathbf{D}_a^{(\gamma)}) v_{[k+1]} := G\, v_{[k+1]}. \tag{5.53}$$

The matrix P is then a unimodular completion of the matrix F of the system (5.21) with inverse G.

Remark 5.3.3 *If the case where the zero-space decomposition is considered, the inverse of the equation* (5.45) *is given by:*

$$z_{[i+1]} = Z_{[i]}^{\dagger L} F_{1,[i-1]} \dots F_{1,[0]} v_{[0]} = \widehat{P}_{[i]} v_{[0]} \tag{5.54}$$

where $Z_{[i]}^{\dagger L}$ *is obtained by the unique following conditions:*

$$Z_{[i]}^{\dagger L} Z_{[i]} = I, \quad Z_{[i]}^{\dagger L} F_{1,[i]}^{\dagger R} = 0 \quad \textit{and} \quad Z_{[i]}^{\dagger L} \widetilde{F}_{1,[i]}^{\perp R} = 0. \tag{5.55}$$

Finally, equations (5.52) and (5.54) constitute the fractional flat output.

Remark 5.3.4 *In the case where the system is fractionally* (-1)*-flat, the same algorithm can be applied on the implicit system* $\widetilde{F}x = 0$.

5.3.3 Fractionally direct flat representation

Inspired by the work presented in Fritzsche et al. (2016b), the algorithm for calculating the unimodular completion can be reduced under the following condition:

Proposition 5.3.3 *Let* $F \in \mathbb{R}[\mathbf{D}_a^\gamma]^{n\times(n+m)}$ *of the system* (5.21) *be hyper-regular. If there exists a column permutation matrix* Π *such that*

$$\widehat{F} \triangleq F\Pi = \begin{pmatrix} S[\mathbf{D}_a^\gamma] & T[\mathbf{D}_a^\gamma] \end{pmatrix} \tag{5.56}$$

with $S[\mathbf{D}_a^\gamma] \in GL_n(\mathbb{R}[\mathbf{D}_a^\gamma])$ *is unimodular, then a unimodular completion of* $\widehat{F}$ *is given by* $\widehat{P} = \begin{pmatrix} 0_{m\times(n-m)} & I_m \end{pmatrix}$ *and a unimodular completion of* F *is given by*

$$P = \widehat{P}\,\Pi^T \tag{5.57}$$

By this way, the vector z *such that* $z = P \begin{pmatrix} x \\ u \end{pmatrix}$ *is a fractional flat output.*

From the expression of P in (5.57), we can see that the m components of z are simply a permutation of m elements of the state and the input vectors. From here,

expression (5.57) is called fractionally direct flat representation and z is a fractional direct flat output.

Remark 5.3.5 *Proposition 5.3.3 is also applicable on the matrix $\widetilde{F}$ in the case of fractionally (-1)-flat system.*

5.4 Application to an academic example

In this section, we apply the unimodular completion algorithm to an academic example (Victor, 2010). The fractional flat output z of the system is computed. In addition, state and input variables are deduced as a function of z and its successive fractional derivatives. Moreover, we use the flatness property to generate a reference trajectory of the inputs and outputs of the system.

Consider the following system

$$\begin{cases} x_1^{(2\nu)} + x_1 - x_2 = u \\ x_2^{(2\nu)} + x_2 - x_1 = 0 \end{cases} \tag{5.58}$$

where $x = (x_1, x_2)^T \in (\mathfrak{H}_0)^2$ is the pseudo-state vector and $u \in \mathfrak{H}_0$ is the input vector, with

$$\mathfrak{H}_0 \triangleq \{f : \mathbb{R} \mapsto \mathbb{R} \mid f \in \mathcal{C}^\infty([0, +\infty[), f(t) = 0, \forall t \leq 0\}. \tag{5.59}$$

We also suppose that the output of the system is given by $y = x_2 \in \mathfrak{H}_0$. In the following, the fractional derivative $\mathbf{D}_0^{(2\nu)}$ will be denoted by $\mathbf{D}^{(2\nu)}$.

The system (5.58) can be represented by

$$Fv_{[0]} = 0 \tag{5.60}$$

with

$$F = \begin{pmatrix} \mathbf{D}^{(2\nu)} + 1 & -1 & -1 \\ -1 & \mathbf{D}^{(2\nu)} + 1 & 0 \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{2\nu}]^{2\times 3} \tag{5.61}$$

and $v_{[0]} = \begin{pmatrix} x \\ \\ u \end{pmatrix}$.

5.4.1 Computation via unimodular completion algorithm

Using expression (5.30), the matrix F given in (5.61) is decomposed into the form

$$F = F_{0,[0]} + F_{1,[0]}\mathbf{D}^{(2\nu)} \tag{5.62}$$

with

$$F_{0,[0]} = \begin{pmatrix} 1 & -1 & -1 \\ \\ -1 & 1 & 0 \end{pmatrix} \quad \text{and} \quad F_{1,[0]} = \begin{pmatrix} 1 & 0 & 0 \\ \\ 0 & 1 & 0 \end{pmatrix}. \tag{5.63}$$

By following the algorithm, first we compute the right pseudo-inverse and the right orthonormal of $F_{1,[0]}$:

$$F_{1,[0]}^{\dagger R} = \begin{pmatrix} 1 & 0 \\ \\ 0 & 1 \\ \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad F_{1,[0]}^{\perp R} = \begin{pmatrix} 0 \\ \\ 0 \\ \\ 1 \end{pmatrix}. \tag{5.64}$$

Then, the first change of coordinates is given by:

$$v_{[0]} = F_{1,[0]}^{\dagger R} v_{[1]} + F_{1,[0]}^{\perp R} w_{[1]} \tag{5.65}$$

with $v_{[1]} = \begin{pmatrix} x_1 \\ \\ x_2 \end{pmatrix}$ and $w_{[1]} = u$, and the system (5.60) becomes:

$$v_{[1]}^{(2\nu)} + \mathfrak{A}_{[0]} v_{[1]} + \mathfrak{B}_{[0]} w_{[1]} = 0 \tag{5.66}$$

with

$$\mathfrak{A}_{[0]} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad \text{and} \quad \mathfrak{B}_{[0]} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \tag{5.67}$$

see equation (5.35).

The matrix $\mathfrak{B}_{[0]}$ is of full column rank but not of full row rank, then we need to reduce the dimension of the system by applying the elimination step. So, for the iteration $i = 1$, we have:

$$F_{1,[1]} = \mathfrak{B}_{[0]}^{\perp L} = \begin{pmatrix} 0 & 1 \end{pmatrix} \quad \text{and} \quad F_{0,[1]} = \mathfrak{B}_{[0]}^{\perp L}\mathfrak{A}_{[0]} = \begin{pmatrix} -1 & 1 \end{pmatrix}. \tag{5.68}$$

Then, we compute

$$F_{1,[1]}^{\dagger R} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{and} \quad F_{1,[1]}^{\perp R} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \tag{5.69}$$

which gives $\mathfrak{A}_{[1]} = 1$ and $\mathfrak{B}_{[1]} = -1$. Here $\mathfrak{B}_{[1]}$ reaches a full row rank and the algorithm ends at this step.

Using (5.52), the unimodular completion of the matrix F is then given by:

$$P = F_{1,[1]}F_{1,[0]} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \tag{5.70}$$

and the system (5.58) is fractionally flat with

$$z = P \begin{pmatrix} x \\ u \end{pmatrix} = x_2 \tag{5.71}$$

as a fractional flat output. Conversely, we can express x_1 and u in function of the flat output $z = x_2$ using (5.53):

$$\begin{pmatrix} x_1 \\ x_2 \\ u \end{pmatrix} = G_{[0]}(\mathbf{D}^{(2\nu)})G_{[1]}(\mathbf{D}^{(2\nu)})z \tag{5.72}$$

with $G_{[i]}(\mathbf{D}^{(2\nu)})$ given by (5.51). Then

$$\begin{pmatrix} x_1 \\ x_2 \\ u \end{pmatrix} = \begin{pmatrix} \mathbf{D}^{(2\nu)} + 1 \\ 1 \\ \mathbf{D}^{(4\nu)} + 2\mathbf{D}^{(2\nu)} \end{pmatrix} z \tag{5.73}$$

which leads to $x_1 = z^{(2\nu)} + z$, $x_2 = z$ and $u = z^{(4\nu)} + 2z^{(2\nu)}$.

Remark 5.4.1 *The same algorithm can be applied on the implicit form of* (5.58)*, because of the hyper-regularity of the matrix* $B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$.

5.4.2 Application on trajectory planning

Suppose that we want the output of the system to move from $y_i = 0$ at $t_i = 0s$ to $y_f = 5$ at $t_f = 30s$, under the condition that these two points are equilibrium points:

$$\begin{aligned} \dot{y}_i &= 0, \qquad \dot{y}_f = 0, \\ \ddot{y}_i &= 0, \qquad \ddot{y}_f = 0. \end{aligned} \tag{5.74}$$

The reference trajectory of the flat output $z = x_2 = y$ is then computed using a fifth-order polynomial interpolation:

$$z_{ref}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5. \tag{5.75}$$

Reference trajectories of x_1 and u are then deduced by flatness using the differential flat equation (5.73). Figure 5.2 illustrates the reference trajectories of the flat output z and of the state x_2 and the input u.

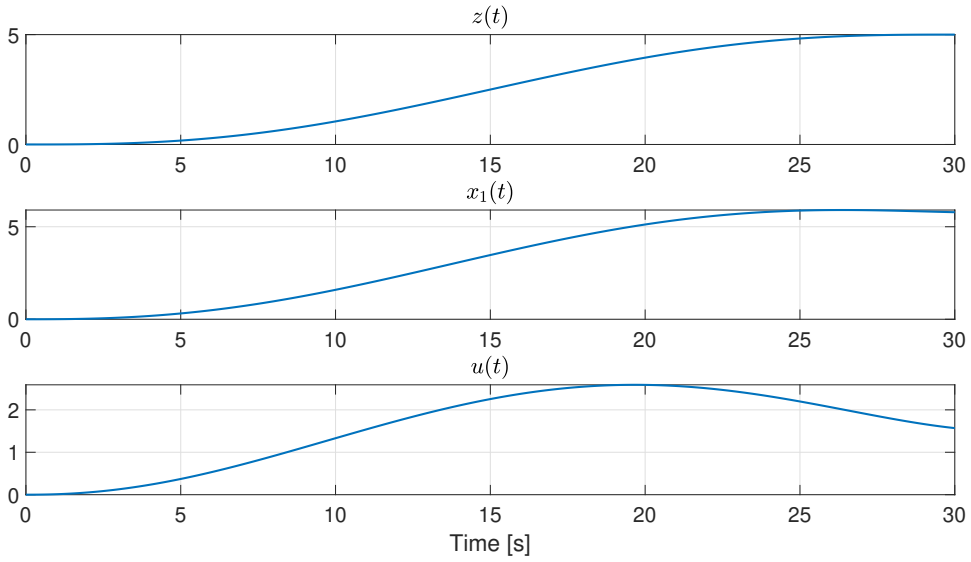

FIGURE 5.2: *Reference trajectories of the system variables*

Finally, Figure 5.3 shows that, by applying the input u calculated by flatness on the system (5.58), the system state x_1 and output x_2 successfully follow their reference trajectories in the absence of disturbances.

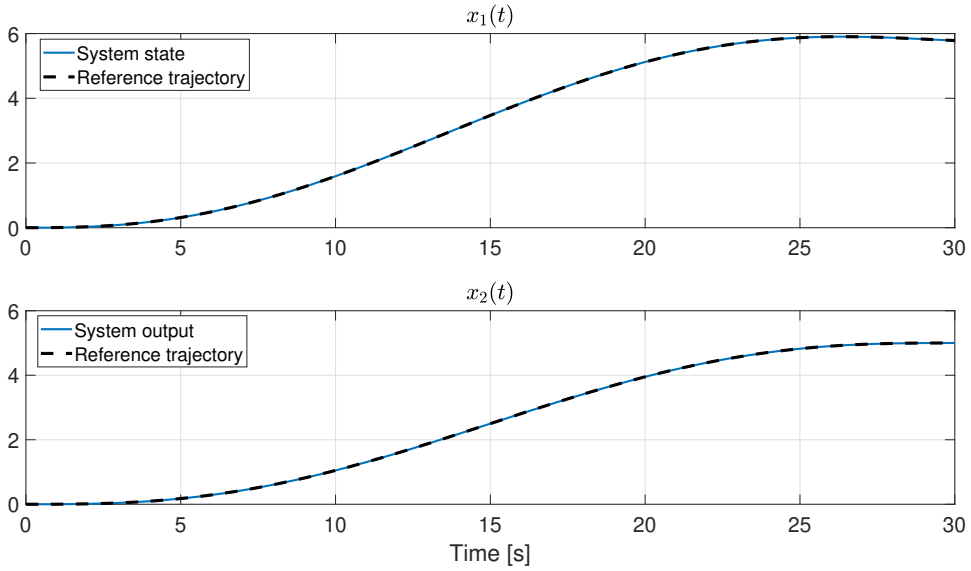

FIGURE 5.3: *System state and output vs. reference trajectories*

5.5 Conclusion

The problem of planning and tracking of trajectory has been longstanding in automatic control. It can be solved by several methods. The differential flatness property of dynamic systems provides a simple way to solve these two problematic issues. Since, in a flat system, all the system inputs, states and outputs can be expressed in function of the flat output and its successive time derivatives, it is sufficient to build a reference trajectory for this flat output, and then deduce those of the inputs, states and outputs, without the need to integrate any differential equation.

The differential flatness was first introduced for the class of rational or integer order systems, and then extended to the class of fractional order linear systems. For the class of fractional order nonlinear systems, this problem is still open and needs some developments in the future.

The main goal of the differential flatness is to find the expression of the flat outputs. Then, after recalling the differential flatness of fractional linear systems, a new method of computation of fractional flat outputs has been introduced, based on the extension of the unimodular completion algorithm, initially developed for the class of integer order systems. This algorithm was then applied on an academic example of fractional linear system. Moreover, the calculated fractional flat output was used for generating reference trajectories of the system.

The flatness of dynamic systems has other applications. It can be used for FTC and FDI as presented in Chapter 3. In the next chapter, an extension of the flatness-based FDI method, introduced by Martínez-Torres et al. (2014), to the class of fractional linear systems is developed with an application on a thermal system.

Chapter 6

Fractionally Differential Flatness for Fault Detection and Isolation

Abstract

This chapter presents an implementation of the extension of the flatness-based FDI method, presented in Chapter 3, to the class of fractional linear flat systems. The fractional flat outputs are used to construct analytical redundant signals, such as redundant outputs and redundant inputs, in order to compare the real measurements of sensors and actuators to their redundancies. Then, due to the presence of noise on system sensors and actuators, the resulting residues are compared to fixed thresholds. Definitions of detectability, isolability and independence between different flat outputs, introduced in Chapter 3, remain the same for the class of fractional linear flat systems. Finally, this method is applied on a fractional linear system, the thermal bi-dimensional system.

6.1 Introduction

In the last decade, many studies have shown that there exist applications like thermal systems (Battaglia et al., 2000), viscoelastic systems (Moreau, Ramus-Serment, and Oustaloup, 2002) and nuclear magnetic resonance systems (Magin et al., 2008) that can be modeled by fractional differential equations. Therefore, it is necessary to develop FDI methods taking advantage of the characteristics of these processes (Aoun et al., 2011).

A first development of fractional models in the diagnosis field was initiated in Aoun et al. (2010). This method is an extension of the dynamic parity space FDI method. See also Aoun et al. (2011), Aribi et al. (2012b), and Aribi et al. (2012a). Later on in Aribi et al. (2013), a scheme of FDI is extended to diagnose fractional order systems. It is based on a bank of fractional unknown input observers.

The characterization of the flat outputs for fractional linear systems has been discussed in the previous chapter. Here we show how these results can be used for FDI in the case of fractional linear flat systems. As in the case of integer-order nonlinear systems, the redundant components are calculated analytically using the fractional flat output. Then, the residual signal is generated by the difference between the real measurements and their redundancies. Definitions of signature matrix, detectability and isolability, introduced in Chapter 3, are also extended to deal with this class of fractional linear systems. The efficiency of this method is proved by simulations on the thermal bi-dimensional system.

6.2 Fractional flatness-based residual generation

Consider the following linear system of fractional order ν:

$$\begin{cases} x^{(\nu)}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{6.1}$$

where $x \in \mathfrak{H}_a^n$ is the pseudo-state vector, $u \in \mathfrak{H}_a^m$ the input vector, $y \in \mathfrak{H}_a^p$ the output vector of dimension $p \geq m$, and A, B, C and D are real matrices with proper

dimensions. Recall that the space $\mathfrak{H}_a$ is the space of causal functions defined by:

$$\mathfrak{H}_a \triangleq \{f : \mathbb{R} \mapsto \mathbb{R} \mid f \in \mathcal{C}^\infty([a, +\infty[), f(t) = 0, \forall t \leq a\}. \tag{6.2}$$

We suppose that the system (6.1) is flat with $z = (z_1, \ldots, z_m) \in \mathfrak{H}_a^m$ as a fractional flat output. Then, the expressions of the state, input and output vectors are linear combinations of the fractional flat output z and its successive fractional derivatives:

$$x_i = \sum_{j=1}^{m} \sum_{k=1}^{\alpha_j} a_{i,j,k} z_j^{(k\nu)} \qquad i = 1, \ldots, n \tag{6.3}$$

$$u_l = \sum_{j=1}^{m} \sum_{k=1}^{\alpha_j+1} b_{l,j,k} z_j^{(k\nu)} \qquad l = 1, \ldots, m \tag{6.4}$$

$$y_q = \sum_{j=1}^{m} \sum_{k=1}^{\beta_j} c_{q,j,k} z_j^{(k\nu)} \qquad q = 1, \ldots, p. \tag{6.5}$$

In turn, the fractional flat output z is a linear combination of x, u and successive fractional derivatives of u:

$$z = h(x, u, u^{(\nu)}, \ldots, u^{(r\nu)}). \tag{6.6}$$

As in Chapter 3, we suppose that the components $y_1, \ldots, y_p$ of the output y are measured by sensors $\mathsf{S}_1, \ldots, \mathsf{S}_p$, respectively, and we denote their measurements by:

$$y^s = (y_1^s, \ldots, y_p^s). \tag{6.7}$$

The values of the input components $u_1, \ldots, u_m$, corresponding to the actuators $\mathsf{A}_1, \ldots, \mathsf{A}_m$, are assumed to be available at every time.

The redundant input vector, denoted by u^z, and the redundant output vector, denoted by y^z, are calculated by flatness using expressions (6.4) and (6.5), respectively. For this reason, a necessary condition for this method to be applicable is that the measurement of the components z_i, for $i = 1, \ldots, m$, of the fractional flat output must be available at every time. These components can be measured by sensors, *i.e.* $z^s = \mathrm{pr}_{\mathbb{R}^m}(y^s)$, or deduced from the measurements of x and u using (6.6), if

these latter measurements are also available. Then, residual signals are defined by the difference between the available measurements and their redundancies:

Definition 6.2.1 *The k^{th}-sensor residue R_{S_k} and l^{th}-input residue R_{A_l}, for $k = 1, \ldots, p$ and $l = 1, \ldots, m$, are given by:*

$$R_{S_k} = y_k^s - y_k^z, \quad \text{and} \quad R_{A_l} = u_l - u_l^z. \tag{6.8}$$

respectively.

In total, we have $p + m$ residues and we denote by r the full vector of residues:

$$r = (r_{S_1}, \ldots, r_{S_p}, r_{A_1}, \ldots, r_{A_m}). \tag{6.9}$$

In the following, we denote by

$$\zeta = (y_1^s, \ldots, y_p^s, u_1, \ldots, u_m) \tag{6.10}$$

the vector of dimension $p + m$ of the available measurements.

Remark 6.2.1 *In order to compute the fractional derivative of the measurements of the fractional flat output, we use the function **dn** of the toolbox CRONE (see Oustaloup et al. (2000)). However, this function is an approximation of the fractional derivative, which does not prevent the existence of a computation error between the exact value of the fractional derivative and the value calculated by the function **dn**.*

6.3 Fractional flatness-based FDI

For the aim of detecting and isolating faults on sensors and actuators, we introduce the following fractional signature matrix:

Definition 6.3.1 (Fractional signature matrix) *Given the vector of residues r defined in* (6.9) *and ζ the vector of available measurements, defined in* (6.10)*, we define by the fractional signature matrix, associated to the fractional flat output z, the matrix* **S** *given by:*

$$\mathbf{S} = \begin{array}{c} \\ r_1 \\ \vdots \\ r_{p+m} \end{array} \begin{array}{c} \begin{array}{cccc} \zeta_1 & \zeta_2 & \dots & \zeta_{p+m} \end{array} \\ \begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} & \dots & \sigma_{1,p+m} \\ \vdots & \vdots & \dots & \vdots \\ \sigma_{p+m,1} & \sigma_{p+m,2} & \dots & \sigma_{p+m,p+m} \end{pmatrix} \end{array} \tag{6.11}$$

with

$$\sigma_{i,j} \triangleq \begin{cases} 0 \quad \text{if} \quad \frac{\partial r_i}{\partial \zeta_j^{(\varrho\nu)}} = 0 \quad \forall \varrho \in \{0,1,\dots\} \\ 1 \quad \text{if} \quad \exists\, \varrho \in \{0,1,\dots\} \text{ s.t. } \frac{\partial r_i}{\partial \zeta_j^{(\varrho\nu)}} \neq 0 \end{cases} . \tag{6.12}$$

A column Σ_j, for $j = 1, \dots, p+m$, of the signature matrix **S** indicates whether a residue r_i is or is not functionally affected by a fault on the measurement ζ_j. So in (6.12), $\sigma_{i,j} = 0$ means that the residue r_i is not affected by a fault on the measurement ζ_j and $\sigma_{i,j} = 1$ otherwise.

Definition 6.3.2 (Fault alarm signature) *A column Σ_j of the signature matrix* **S** *is called fault alarm signature, associated to the sensor/actuator ζ_j.*

The following definitions of *detectability* and *isolability*, in the fractional flatness context, are the same as the definitions presented in Chapter 3:

Definition 6.3.3 (Detectability) *A fault on a sensor/actuator ζ_j is detectable if, and only if there exists at least one $i \in \{1, \dots, p\}$ such that $\sigma_{i,j} = 1$.*

Definition 6.3.4 (Isolability) *A fault on a sensor S_k, $k = 1, \dots, p$, is said isolable if, and only if, its corresponding fault alarm signature Σ_k in the signature matrix* **S** *is distinct from the others,* i.e.

$$\Sigma_k \neq \Sigma_j, \quad \forall j = 1, \dots, p+m,\ j \neq k. \tag{6.13}$$

An isolable fault on the actuator A_l, for $l = 1, \dots, m$, is defined analogously:

$$\Sigma_{p+l} \neq \Sigma_j, \quad \forall j = 1, \dots, p+m,\ j \neq p+l. \tag{6.14}$$

Then, from Definition 6.3.4, if the signature matrix $\mathbf{S}$ has two identical signatures, *i.e.* $\Sigma_i = \Sigma_j$, for two different sensors/actuators $\zeta_i \neq \zeta_j$, then we cannot make a decision on the faulty device. Accordingly, we define by μ the number of distinct fault alarm signatures of the signature matrix $\mathbf{S}$ associated to z, *i.e.* μ is the number of isolable faults associated to z.

Therefore, the full isolability of faults is given by the following proposition:

Proposition 6.3.1 *A full isolability of faults is achieved if, and only if, the signature matrix* $\mathbf{S}$ *has* $p + m$ *distinct fault alarm signatures,* i.e. $\mu = p + m$.

Remark 6.3.1 *If, by using a single fractional flat output* z*, the full isolability of faults is not achieved,* i.e. *the number of distinct fault alarm signatures* $\mu < p + m$*, then we need to find another fractional flat output vector which is independent of* z*. The condition of independence between two different fractional flat output vectors is the same as the condition given in Definition 3.5.2. That is, two fractional flat outputs are independent if, and only if, by using them together, the number of isolable faults* $\widetilde{\mu}$ *of the augmented signature matrix* $\widetilde{\mathbf{S}}$*, increases,* i.e. $\widetilde{\mu} > \mu$.

In the next section, the proposed fractional flatness-based FDI method is applied on the thermal bi-dimensional system. Moreover, the effectiveness of this method is proved by simulations.

6.4 Thermal bi-dimensional system

Thermal systems are those that involve the storage and transfer of heat. Examples of thermal systems are solar thermal systems, radiators, electric stove, among many others. The thermal system paradigm is a popular example in automatic control. It has been used as a benchmark for designing controllers, trajectory planning and system diagnostic. In this section, we apply the fractional flatness-based FDI method to the thermal bi-dimensional system.

6.4.1 System description

The thermal bi-dimensional system is about a 2D metallic sheet which is isolated, *i.e.* without heat losses (see Figure 6.1). The variable $T(x, y, t)$ represents the temperature at a point (x, y) at time t. The temperature is controlled by the heat flux $\varphi(t)$, applied at the point $(0, 0)$.

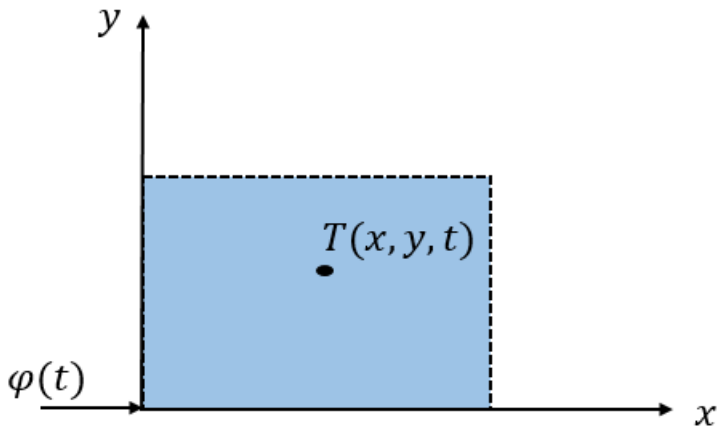

FIGURE 6.1: *Thermal bi-dimensional System*

The heated metallic model is represented by the following heat equation:

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} - \frac{1}{\alpha}\frac{\partial}{\partial t}\right) T(x, y, t) = 0 \tag{6.15}$$

where α is the coefficient of diffusivity. In polar coordinates:

$$x = \rho\cos(\theta), \qquad y = \rho\sin(\theta), \tag{6.16}$$

equation (6.15) becomes:

$$\frac{1}{\alpha}\frac{\partial T(\rho, \theta, t)}{\partial t} = \frac{\partial^2 T(\rho, \theta, t)}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial T(\rho, \theta, t)}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 T(\rho, \theta, t)}{\partial^2 \theta}. \tag{6.17}$$

On the thermal bi-dimensional system, we consider the following conditions:

– the boundary conditions:

$$-\lambda \lim_{\rho \to 0} \frac{\pi}{2}\rho\frac{\partial T(\rho, \theta, t)}{\partial \rho} = \varphi(t) \qquad \forall t > 0 \tag{6.18}$$

and

$$\frac{\partial T(\rho,0,t)}{\partial \rho}=0 \quad \text{and} \quad \frac{\partial T(\rho,\frac{\pi}{2},t)}{\partial \rho}=0, \qquad \forall \rho>0 \tag{6.19}$$

where λ is the coefficient of conductivity

– the limit condition:

$$\lim_{\rho\to\infty} T(\rho,\theta,t)=0 \qquad \forall\, \theta \in \left[0,\frac{\pi}{2}\right], \quad \forall t>0 \tag{6.20}$$

– the initial condition known as Cauchy condition:

$$T(\rho,\theta,0)=0 \qquad \forall \rho>0, \quad \forall \theta \in \left[0,\frac{\pi}{2}\right]. \tag{6.21}$$

A differential flatness method for resolving the bi-dimensional thermal system has been developed in Belghith et al., 2003. In this method, a Fourier transformation is applied to a space variable, which transforms the bi-dimensional system into a mono-dimensional one. Then, the problem of mono-dimensional control has been solved using the Laplace transformation of the variable t and the differential flatness. In this thesis, a solution for the system is determined using the Laplace transformation of the equation (6.17) and the method of separation of variables. The solution is a modified Bessel function of second kind of order 0. This method leads to polynomials in s^{ν}, where $s \in \mathbb{C}$ is the Laplace variable, in explicit form, which are well adapted for pseudo-state representation (6.1).

Hypothesis: The temperature is assumed to be $0^{\circ}C$ at every point of the metallic sheet for all $t \leq 0$.

The Laplace transformation of the equation (6.17) is given by:

$$\frac{s}{\alpha}\widehat{T}(\rho,\theta,s)=\frac{\partial^2 \widehat{T}(\rho,\theta,s)}{\partial \rho^2}+\frac{1}{\rho}\frac{\partial \widehat{T}(\rho,\theta,s)}{\partial \rho}+\frac{1}{\rho^2}\frac{\partial^2 \widehat{T}(\rho,\theta,s)}{\partial^2 \theta} \tag{6.22}$$

where

$$\widehat{T}(\rho,\theta,s)=\int_0^{+\infty}T(\rho,\theta,t)e^{-st}dt \tag{6.23}$$

is the Laplace transformation of $T(\rho,\theta,t)$. Using the separation of variables method, the temperature $\widehat{T}(\rho,\theta,s)$ can be written into the form

$$\widehat{T}(\rho,\theta,s)=\widehat{T}_\rho(\rho,s)\widehat{T}_\theta(\theta,s) \tag{6.24}$$

where $\widehat{T}_\rho(\rho,s)$ is function of ρ and s and $\widehat{T}_\theta(\theta,s)$ is function of θ and s. By injecting (6.24) in (6.22), we get

$$\frac{s}{\alpha}\widehat{T}_\rho(\rho,s)\widehat{T}_\theta(\theta,s)=\widehat{T}_\theta(\theta,s)\frac{\partial^2\widehat{T}_\rho(\rho,s)}{\partial\rho^2}+\frac{1}{\rho}\widehat{T}_\theta(\theta,s)\frac{\partial\widehat{T}_\rho(\rho,s)}{\partial\rho}+\frac{1}{\rho^2}\widehat{T}_\rho(\rho,s)\frac{\partial^2\widehat{T}_\theta(\theta,s)}{\partial\theta^2}. \tag{6.25}$$

Multiplying by $\dfrac{\rho^2}{\widehat{T}(\rho,\theta,s)}$, equation (6.25) becomes

$$\frac{\rho^2}{\widehat{T}_\rho(r\rho,s)}\frac{\partial^2\widehat{T}_\rho(\rho,s)}{\partial\rho^2}+\frac{\rho}{\widehat{T}_\rho(\rho,s)}\frac{\partial\widehat{T}_\rho(\rho,s)}{\partial\rho}-\frac{\rho^2 s}{\alpha}+\frac{1}{\widehat{T}_\theta(\theta,s)}\frac{\partial^2\widehat{T}_\theta(\theta,s)}{\partial\theta^2}=0. \tag{6.26}$$

Equation (6.26) is divided into two equations, the first one is in function of the variable ρ, and the second one is in function of the variable θ, then it is clear that each equation is a constant:

$$\frac{\rho^2}{\widehat{T}_\rho(\rho,s)}\frac{\partial^2\widehat{T}_\rho(\rho,s)}{\partial\rho^2}+\frac{\rho}{\widehat{T}_\rho(\rho,s)}\frac{\partial\widehat{T}_\rho(\rho,s)}{\partial\rho}-\frac{\rho^2 s}{\alpha}=m^2 \tag{6.27}$$

$$\frac{1}{\widehat{T}_\theta(\theta,s)}\frac{\partial^2\widehat{T}_\theta(\theta,s)}{\partial\theta^2}=-m^2 \tag{6.28}$$

Remark 6.4.1 *Because of the symmetry of the metallic sheet, the temperature $\widehat{T}_\theta(\theta,s)$ is constant with respect to θ,* i.e. *$\widehat{T}_\theta(\theta,s)=A_1(s)$, and then $m=0$.*

Equation (6.27) is a modified Bessel equation:

$$\rho^2\frac{\partial^2\widehat{T}_\rho(\rho,s)}{\partial\rho^2}+\rho\frac{\partial\widehat{T}_\rho(\rho,s)}{\partial\rho}-\frac{s\rho^2}{\alpha}\widehat{T}_\rho(\rho,s)=0 \tag{6.29}$$

its solution is a modified Bessel function of the form:

$$\widehat{T}_\rho(\rho,s) = B_1(s)I_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big) + B_2(s)K_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big). \tag{6.30}$$

The modified Bessel function of the first kind I_0 is an exponentially growing function, then, according to the limit condition (6.20), $B_1(s) = 0$, and hence:

$$\widehat{T}_\rho(\rho,s) = B_2(s)K_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big). \tag{6.31}$$

Finally, the solution of the thermal bi-dimensional system is given by:

$$\widehat{T}(\rho,\theta,s) = A_1(s)\,B_2(s)K_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big). \tag{6.32}$$

According to the boundary condition, the heat flux is given by:

$$\begin{aligned}\widehat{\varphi}(s) &= \lambda\lim_{\rho\to 0}\frac{\pi}{2}A_1(s)B_2(s)\,\rho\sqrt{\frac{s}{\alpha}}K_1\Big(\rho\sqrt{\frac{s}{\alpha}}\Big)\\ &= \lim_{\rho\to 0}\lambda\frac{\pi}{2}A_1(s)B_2(s)\,\rho\sqrt{\frac{s}{\alpha}}\frac{1}{\rho\sqrt{\frac{s}{\alpha}}}\\ &= \lambda\frac{\pi}{2}A_1(s)B_2(s)\end{aligned} \tag{6.33}$$

since $\dfrac{\partial K_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big)}{\partial\rho} = -\sqrt{\frac{s}{\alpha}}K_1\Big(\rho\sqrt{\frac{s}{\alpha}}\Big)$ and $K_1\Big(\rho\sqrt{\frac{s}{\alpha}}\Big) \sim \frac{1}{\rho\sqrt{\frac{s}{\alpha}}}$ in the neighbourhood of 0.

The transfer function of the system, called the thermal impedance, is defined by

$$\widehat{H}(\rho,\theta,s) = \frac{\widehat{T}(\rho,\theta,s)}{\widehat{\varphi}(s)} = \frac{A_1(s)B_2(s)K_0\Big(\rho\sqrt{\frac{s}{\alpha}}\Big)}{\lambda\frac{\pi}{2}A_1(s)B_2(s)} \tag{6.34}$$

which, for ρ sufficiently large, is equivalent to:

$$\widehat{H}(\rho,\theta,s) = \frac{2\sqrt{2\pi}}{\lambda\pi}\frac{1}{\sqrt{\rho\sqrt{\frac{s}{\alpha}}}}e^{-\rho\sqrt{\frac{s}{\alpha}}}. \tag{6.35}$$

Applying the Padé approximation at the order **K** of the pure delay (Baker et al., 1996) at a point (x_0, y_0), which corresponds to (ρ_0, θ_0) with $\rho_0 = \sqrt{x_0^2 + y_0^2}$, gives:

$$\widehat{H}_{\mathbf{K}}(\rho_0, \theta_0, s) \approx \frac{\sqrt{2\pi\sqrt{\alpha}}}{\lambda\pi\sqrt{\rho_0}} \frac{\sum_{k=0}^{\mathbf{K}} (-1)^k C'_k s^{\frac{k}{2}}}{\sum_{k=0}^{\mathbf{K}} C'_k s^{\frac{2k+1}{4}}} \tag{6.36}$$

with $C'_k = \dfrac{C_k}{|C_{\mathbf{K}}|}$ and $C_k = \dfrac{(2\mathbf{K}-k)!\mathbf{K}!}{(2\mathbf{K}!)k!(\mathbf{K}-k)!}\left(\dfrac{\rho_0}{\sqrt{\alpha}}\right)^k$.

6.4.1.1 Explicit thermal system

The transfer function $H_{\mathbf{K}}$ of the thermal bi-dimensional system is of fractional order multiple of $\nu = \frac{1}{4}$ and can be written in the form of a pseudo-state representation:

$$X^{(\nu)} = AX + BU, \qquad T_{\mathbf{K}}(x_0, y_0, t) = CX \tag{6.37}$$

with

$$A = \begin{pmatrix} 0 & -C'_{\mathbf{K}-1} & 0 & -C'_{\mathbf{K}-2} & \cdots & 0 & -C'_0 & 0 \\ & & & I_{2\mathbf{K}} & & & & 0_{2\mathbf{K}\times 1} \end{pmatrix} \in \mathbb{R}^{(2\mathbf{K}+1)\times(2\mathbf{K}+1)}, \tag{6.38}$$

$$B = \begin{pmatrix} 1 \\ \\ 0_{2\mathbf{K}\times 1} \end{pmatrix} \in \mathbb{R}^{(2\mathbf{K}+1)\times 1}, \tag{6.39}$$

and

$$C = \frac{\sqrt{2\pi\sqrt{\alpha}}}{\lambda\pi\sqrt{\rho_0}} \begin{pmatrix} (-1)^{\mathbf{K}} C'_{\mathbf{K}} & 0 & \cdots & 0 & -C'_1 & 0 & C'_0 \end{pmatrix} \in \mathbb{R}^{1\times(2\mathbf{K}+1)}. \tag{6.40}$$

The pseudo-state vector X is of size $n = 2\mathbf{K} + 1$ and denoted by

$$X = \begin{pmatrix} X_0 \\ \vdots \\ X_{2\mathbf{K}} \end{pmatrix} \tag{6.41}$$

and the input U is the heat flux, *i.e.* $U = \varphi(t)$. The dimension of the input U and the output $T_{\mathbf{K}}$ are respectively $m = 1$ and $p = 1$. The system (6.37) can also be written into the form

$$\mathcal{A}X = BU, \qquad T_{\mathbf{K}}(x_0, y_0, t) = CX \tag{6.42}$$

where $\mathcal{A}$ is a $\mathbb{R}[\mathbf{D}^{\frac{1}{4}}]$-polynomial matrix of size $(2\mathbf{K}+1) \times (2\mathbf{K}+1)$ and given by

$$\mathcal{A} = \begin{pmatrix} \mathbf{D}^{\frac{1}{4}} & C'_{\mathbf{K}-1} & 0 & \cdots & 0 & C'_0 & 0 \\ -1 & \mathbf{D}^{\frac{1}{4}} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & -1 & \ddots & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & 0 \\ 0 & & \cdots & & 0 & -1 & \mathbf{D}^{\frac{1}{4}} \end{pmatrix}. \tag{6.43}$$

6.4.1.2 Implicit thermal system

The matrix B, being in its Smith form, is hyper-regular. Then, according to Proposition 5.2.2, there exist two matrices $\widetilde{F} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times(2\mathbf{K}+1)}$ and $R \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{1\times(2\mathbf{K}+1)}$ such that

$$\mathcal{A} = \begin{pmatrix} R \\ \widetilde{F} \end{pmatrix} \tag{6.44}$$

with

$$R = \begin{pmatrix} \mathbf{D}^{\frac{1}{4}} & C'_{\mathbf{K}-1} & 0 & \cdots & 0 & C'_0 & 0 \end{pmatrix} \tag{6.45}$$

and

$$\widetilde{F} = \begin{pmatrix} -1 & \mathbf{D}^{\frac{1}{4}} & 0 & \cdots & 0 \\ 0 & -1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & \mathbf{D}^{\frac{1}{4}} \end{pmatrix}. \tag{6.46}$$

Then, the implicit thermal system, associated to (6.37) is given by

$$\begin{cases} RX = U \\ \widetilde{F}X = 0 \end{cases}. \tag{6.47}$$

In the next section, we use the unimodular completion algorithm, presented in Chapter 5, to compute a fractional flat output of the thermal bi-dimensional system.

6.4.2 Computation of the fractional flat output

According to Proposition 5.3.2, a vector $Z = (Z_1, \ldots, Z_m) \in \mathfrak{H}^m$ such that $Z = PX$, with $P \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{m\times 1}$, is a fractional flat output of the implicit bi-dimensional thermal system (6.47) if, and only if, P is a unimodular completion of $\widetilde{F}$.

6.4.2.1 Unimodular completion algorithm

Starting with the iteration $i = 0$ (see (5.30)), the matrix $\widetilde{F}$ can be written in the form

$$\widetilde{F} = \widetilde{F}_{0,[0]} + \widetilde{F}_{1,[0]}\mathbf{D}^{\frac{1}{4}} \tag{6.48}$$

where

$$\widetilde{F}_{0,[0]} = \begin{pmatrix} -1 & 0 & \cdots & \cdots & 0 \\ 0 & -1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & -1 & 0 \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times(2\mathbf{K}+1)} \tag{6.49}$$

and

$$\widetilde{F}_{1,[0]} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & & 0 & 1 \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times(2\mathbf{K}+1)}. \tag{6.50}$$

Then, we compute

$$\mathfrak{A}_{[0]} = \widetilde{F}_{0,[0]}\widetilde{F}^{\dagger R}_{1,[0]} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & -1 & 0 \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times 2\mathbf{K}} \tag{6.51}$$

with $\widetilde{F}^{\dagger R}_{1,[0]}$ is the right pseudo-inverse of $\widetilde{F}_{1,[0]}$, and

$$\mathfrak{B}_{[0]} = \widetilde{F}_{0,[0]}\widetilde{F}^{\perp R}_{1,[0]} = \begin{pmatrix} -1 \\ 0_{(2\mathbf{K}-1)\times 1} \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times 1} \tag{6.52}$$

with $\widetilde{F}^{\perp R}_{1,[0]}$ is the right orthonormal of $\widetilde{F}_{1,[0]}$.

The matrix $\mathfrak{B}_{[0]}$ is always of full column rank but may not be of full row rank,

depending on the value of $\mathbf{K}$. If *rank* $\mathfrak{B}_{[0]} = r < 2\mathbf{K}$, then we move to the Elimination step and we compute

$$\widetilde{F}_{0,[1]} = \mathfrak{B}_{[0]}^{\perp L}\mathfrak{A}_{[0]} \quad \text{and} \quad \widetilde{F}_{1,[1]} = \mathfrak{B}_{[0]}^{\perp L}. \tag{6.53}$$

Then, we apply the Reduction step for iteration $i = 1$. Finally, the computations end at iteration $i = 2\mathbf{K} - 1$ where $\mathfrak{B}_{[2\mathbf{K}-1]}$ reaches full row rank.

The unimodular completion matrix of $\widetilde{F}$ is then given by:

$$P = \widetilde{F}_{1,[2\mathbf{K}-1]} \dots \widetilde{F}_{1,[1]}\widetilde{F}_{1,[0]} = \begin{pmatrix} 0_{1\times 2\mathbf{K}} & 1 \end{pmatrix}, \tag{6.54}$$

and a fractional flat output is given by

$$Z = PX = \begin{pmatrix} 0_{1\times 2\mathbf{K}} & 1 \end{pmatrix} \begin{pmatrix} X_0 \\ \vdots \\ X_{2\mathbf{K}} \end{pmatrix} = X_{2\mathbf{K}}. \tag{6.55}$$

Conversely, the state X can be computed by

$$X = G(\mathbf{D}^{\frac{1}{4}})Z = G_{[0]}(\mathbf{D}^{\frac{1}{4}})G_{[1]}(\mathbf{D}^{\frac{1}{4}}) \dots G_{[2\mathbf{K}-1]}(\mathbf{D}^{\frac{1}{4}})Z \tag{6.56}$$

where $G(\mathbf{D}^{\frac{1}{4}})$ is given by

$$G(\mathbf{D}^{\frac{1}{4}}) = \begin{pmatrix} \mathbf{D}^{\frac{\mathbf{K}}{2}} \\ \mathbf{D}^{\frac{\mathbf{K}}{2}-\frac{1}{4}} \\ \vdots \\ \mathbf{D}^{\frac{1}{4}} \\ 1 \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{(2\mathbf{K}+1)\times 1}. \tag{6.57}$$

Finally, the input U is computed by

$$U = RX = \sum_{k=0}^{\mathbf{K}} C'_k \mathbf{D}^{\frac{2k+1}{4}} Z \tag{6.58}$$

and the output $T_{\mathbf{K}}$ is computed by

$$T_{\mathbf{K}} = CX = \frac{\sqrt{2\pi\sqrt{\alpha}}}{\lambda\pi\sqrt{\rho_0}} \sum_{k=0}^{\mathbf{K}} (-1)^k C'_k \mathbf{D}^{\frac{k}{2}} Z. \tag{6.59}$$

6.4.2.2 Fractionally direct flat output

The unimodular completion algorithm provides a direct way to compute the fractional flat output. See section 5.3.3. In fact, the matrix $\widetilde{F} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{2\mathbf{K}\times(2\mathbf{K}+1)}$ is of the form $\widetilde{F} = (S \;\; T)$ with

$$S = \begin{pmatrix} -1 & \mathbf{D}^{\frac{1}{4}} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \mathbf{D}^{\frac{1}{4}} \\ 0 & \cdots & \cdots & 0 & -1 \end{pmatrix} \in GL_{2\mathbf{K}}(\mathbb{R}[\mathbf{D}^{\frac{1}{4}}]) \tag{6.60}$$

is unimodular and $T = \begin{pmatrix} 0_{(2\mathbf{K}-1)\times 1} \\ \mathbf{D}^{\frac{1}{4}} \end{pmatrix} \in \mathbb{R}[\mathbf{D}^{\frac{1}{4}}]^{(2\mathbf{K}-1)\times 1}$. Then, according to proposition 5.3.3, the system admits a fractionally direct flat representation and a unimodular completion of $\widetilde{F}$ is given by

$$P = \begin{pmatrix} 0_{1\times 2\mathbf{K}} & 1 \end{pmatrix}. \tag{6.61}$$

and we retrieve the fractionally flat output Z

$$Z = PX = \begin{pmatrix} X_0 \\ \vdots \\ X_{2\mathbf{K}} \end{pmatrix} = X_{2\mathbf{K}}. \tag{6.62}$$

6.4.3 Fault detection and isolation

In the thermal bi-dimensional system, we measure the temperature at the point (x_0, y_0) of the metallic sheet. Thus, we have only one sensor at this point, and we denote the measured temperature by $T^s_{\mathbf{K}}(x_0, y_0, t)$. Moreover, there is a single actuator that produces the heat flux $\varphi(t)$.

In order to compute the redundant output and redundant input using the fractional flatness-based method, the measurement of the flat output Z must be available at every time. For the thermal bi-dimensional system, the only available measurements are the output $T^s_{\mathbf{K}}(x_0, y_0, t)$ and the input $\varphi(t)$. However, the measurement of the flat output Z, denoted by Z^s, can be computed from the measurement of the temperature $T^s_{\mathbf{K}}(x_0, y_0, t)$. In fact, from (6.59) we have

$$T_{\mathbf{K}}(x_0, y_0, t) = W(\mathbf{D}^{\frac{1}{4}})Z(t) \tag{6.63}$$

where $W(\mathbf{D}^{\frac{1}{4}})$ is given by

$$W(\mathbf{D}^{\frac{1}{4}}) = \frac{\sqrt{2\pi\sqrt{\alpha}}}{\lambda\pi\sqrt{\rho_0}} \sum_{k=0}^{\mathbf{K}} (-1)^k C'_k \mathbf{D}^{\frac{k}{2}}. \tag{6.64}$$

$W(\mathbf{D}^{\frac{1}{2}})$ is a $(\mathbf{D}^{\frac{1}{2}})$-polynomial, its inverse is given by

$$W_{inv}(\mathbf{D}^{\frac{1}{4}}) = \frac{\lambda\pi\sqrt{\rho_0}}{\sqrt{2\pi\sqrt{\alpha}} \sum_{k=0}^{\mathbf{K}} (-1)^k C'_k \mathbf{D}^{\frac{k}{2}}}. \tag{6.65}$$

Then, the measurement of the fractional flat output at every time is computed by

$$Z^s(t) = W_{inv}(\mathbf{D}^{\frac{1}{4}})T^s_{\mathbf{K}}(x_0, y_0, t). \tag{6.66}$$

The redundant output, denoted by $T^z_{\mathbf{K}}(x_0, y_0, t)$, is computed using (6.5), and the redundant input, denoted by $\varphi^z(t)$, is computed using (6.4). The fractional derivatives of the flat output measurements are computed using the function **dn** of the toolbox CRONE, see Remark 6.2.1. Then, the vector of residues, associated to Z^s, is given by

$$r = \begin{pmatrix} R_{\mathsf{S}} \\ \\ R_{\mathsf{A}} \end{pmatrix} = \begin{pmatrix} T^s_{\mathbf{K}}(x_0, y_0, t) - T^z_{\mathbf{K}}(x_0, y_0, t) \\ \\ \varphi(t) - \varphi^z(t) \end{pmatrix}. \tag{6.67}$$

It is important to note that we work with an open-loop system. Then, only the residue that depends on the measurement of the temperature $T^s_{\mathbf{K}}$ is affected if a fault occurs on the sensor S. Similarly, only the residue that depends on the measurement of the heat flux φ is affected, if a fault occurs on the actuator A. Then, the signature matrix $\mathbf{S}$, associated to the fractional flat output Z^s, is given by:

$$\mathbf{S} = \begin{pmatrix} 1 & 0 \\ \\ 0 & 1 \end{pmatrix}. \tag{6.68}$$

According to Definition 6.3.3, all faults on the system sensor and actuator are detectable, and according to Definition 6.3.4, all faults are isolable. Moreover, since $\mu = 2 = p + m$, we have full isolability of faults using the fractional flat output Z^s.

6.4.4 Simulation Results

The fractional flatness-based FDI method, applied on the thermal bi-dimensional system, is proved by simulations.

6.4.4.1 Trajectory planning

In our simulations, we want the temperature to go from a resting state of temperature $T_0 = 0°C$ to a resting state of temperature $T_f = 30°C$ in a period of 2500s, at the point $x_0 = 0.005m$ and $y_0 = 0.002m$. Then, the initial and final conditions of the temperature are the following:

$$\begin{aligned} &T(x_0, y_0, 0) \triangleq T_0 = 0, \\ &T(x_0, y_0, t_f) \triangleq T_f = 30, \quad t_f = 2500s, \\ &T^{(l)}(x_0, y_0, 0) = 0, \quad l = 1, 2, \\ &T^{(l)}(x_0, y_0, t_f) = 0, \quad l = 1, 2. \end{aligned} \tag{6.69}$$

Then, the desired reference trajectory is calculated by polynomial interpolation of order 5:

$$T_{ref}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5. \tag{6.70}$$

The reference trajectory of the temperature $T_{ref}(t)$ and its derivatives are illustrated in Figure 6.2.

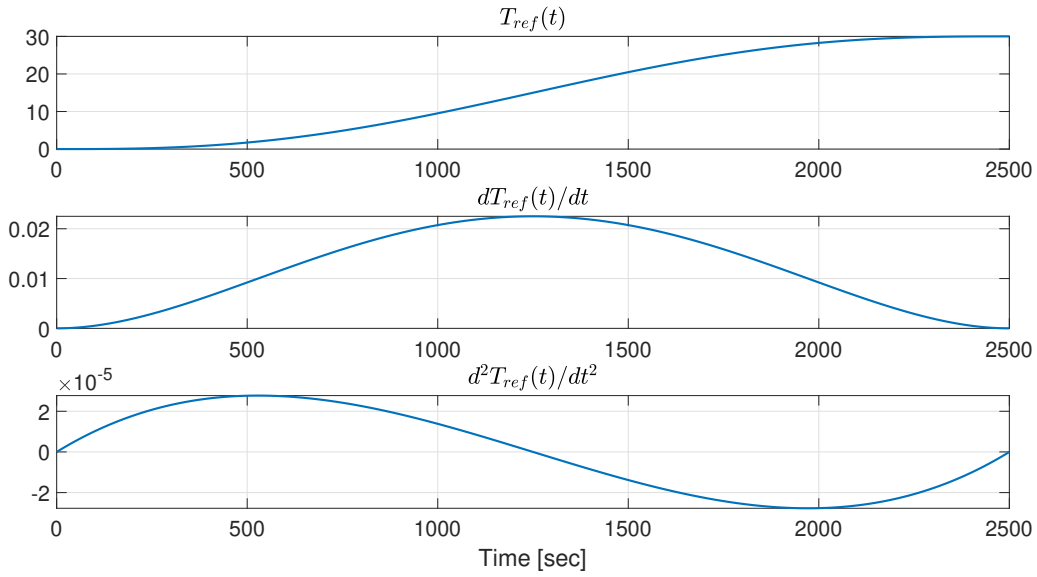

FIGURE 6.2: *Reference trajectory of the temperature $T(x_0, y_0, t)$ and its derivatives*

The reference trajectory of the fractional flat output $Z_{ref}(t)$ is deduced using (6.66) with $\mathbf{K} = 20$, and it is illustrated in Figure 6.3. Since the flat output is a pseudo-state variable, it is difficult to give it a physical sens, this is why there is no unit of measurement associated to $z(t)$.

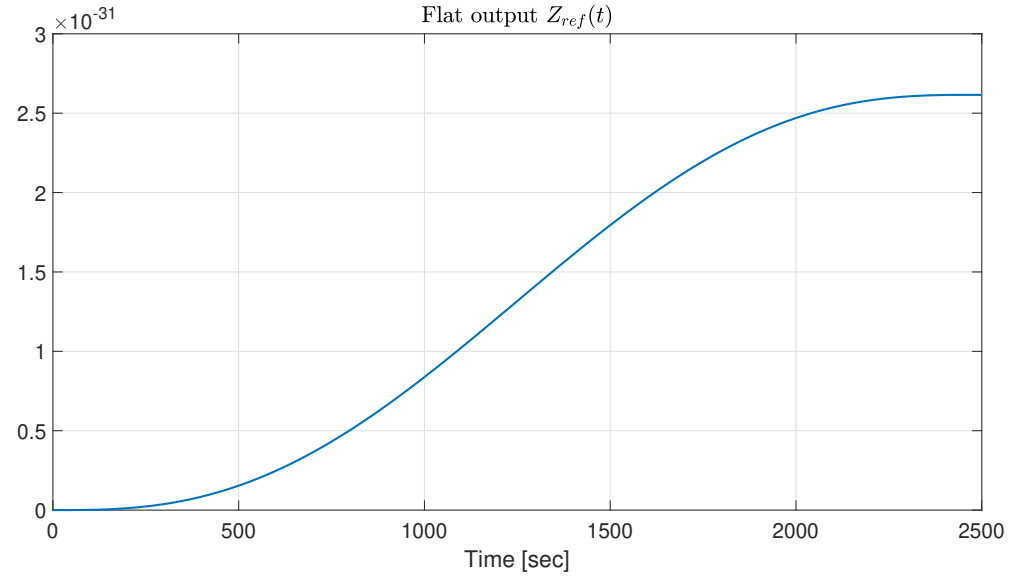

FIGURE 6.3: *Reference trajectory of the flat output* $Z_{ref}(t)$

Finally, a reference trajectory of the heat flux $\varphi_{ref}(t)$ is deduced by flatness using (6.58). See Figure 6.4.

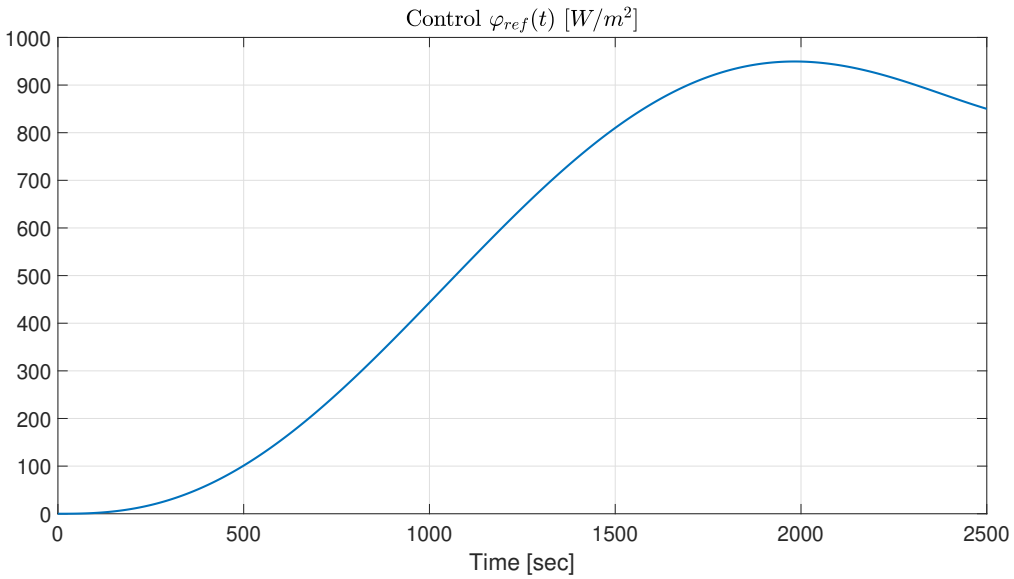

FIGURE 6.4: *Reference trajectory of the heat flux* $\varphi(t)$

Figure 6.5 shows that, by applying the heat flux $\varphi_{ref}(t)$ on the system (6.37), in the absence of perturbations, the output of the system follows its reference trajectory.

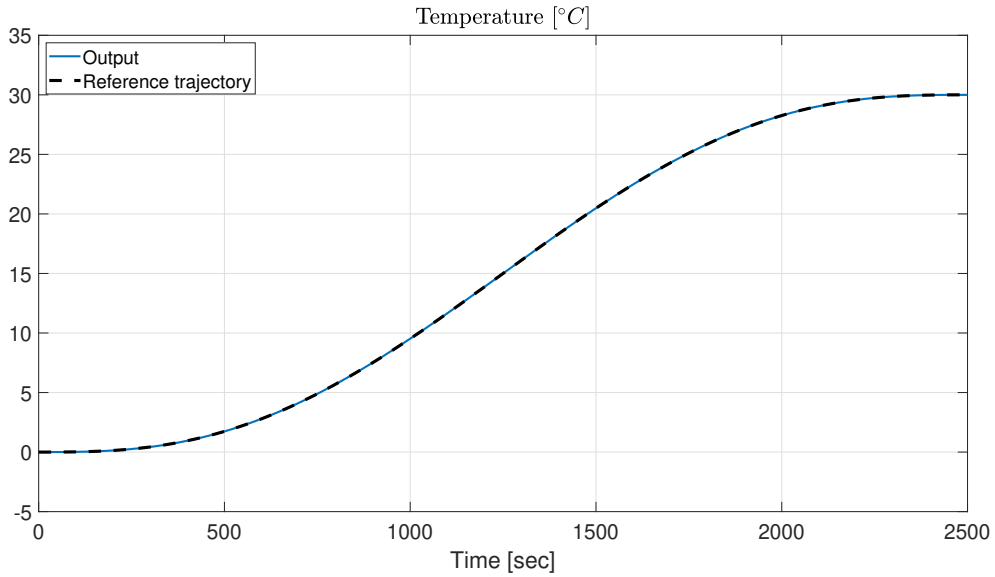

FIGURE 6.5: *Reference trajectory vs. exact solution*

In our simulations, white Gaussian noise is added to the sensor and the actuator with a level corresponding to the actual process level, see Table 6.1.

	Mean	Variance	Power
Sensor	0	0.13	1×10^{-2}
Actuator	0	0.13	1×10^{-2}

TABLE 6.1: *Parameter values of the added white Gaussian noise*

Since there is noise on the system sensor and actuator, a threshold is fixed for each residue. For this purpose, several nominal simulations were realized with different initial and final conditions. The amplitude of the detection threshold is calculated by selecting the worst case among all simulations results, plus a 5% safety margin to avoid false alarms. The values of the maximum and minimum threshold for each residue are given in Table 6.2. In the figures below, the thresholds are normalized between -1 and 1.

	Max	Min
$R_S\ [^\circ C]$	0.0089	-0.0078
$R_A[W/m^2]$	0.0404	-0.1122

TABLE 6.2: *Values of the maximum and minimum threshold for each residue*

Moreover, because of the presence of noise, a low-pass filter of order 4 and cutoff frequency $f_c = 0.2Hz$ is used in order to filter the noise.

In the following, we apply multiplicative and additive faults on the sensor and the actuator.

6.4.4.2 Multiplicative sensor fault

For multiplicative sensor fault, at time $t = 1000s$, the sensor measures only 80% of the temperature, so the actual temperature is 20% above the measurement indicated by the sensor. See Figure 6.6.

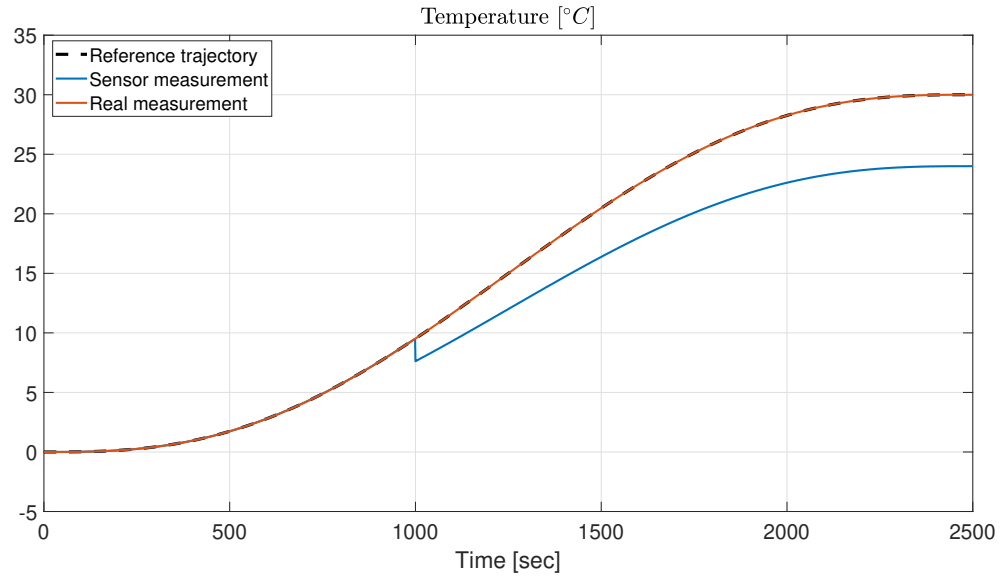

FIGURE 6.6: *Real temperature at point (x_0, y_0) vs. sensor measurement*

On the other hand, since there is no controller on the system, the input $\varphi(t)$ and therefore the system states including the fractional flat output $Z(t)$ are not affected by this fault. The redundant output and the redundant input are then calculated

analytically, using the flat output $Z(t)$. Figure 6.7 shows the sensor measurement and the redundant sensor. We can also see that the redundant sensor or the virtual sensor measures the real temperature. Then, for the aim of fault tolerant control (FTC), the virtual sensor can replace the faulty sensor and the system can operate safely.

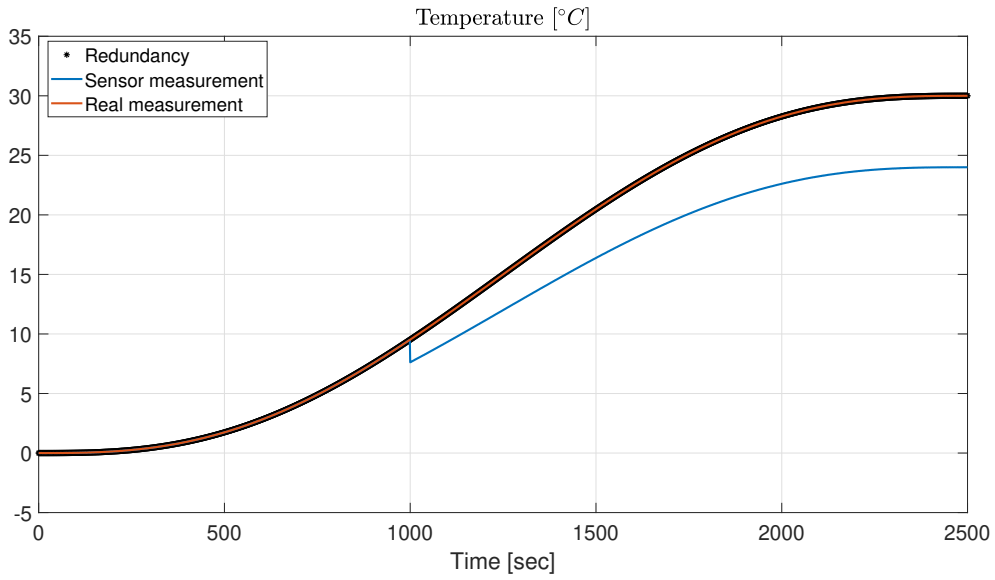

FIGURE 6.7: *Real temperature at point (x_0, y_0) and redundant sensor vs. sensor measurement*

The residue vector associated to the flat output $Z(t)$ is then calculated by the difference between the measurements and their redundancies. In the case of multiplicative fault on the sensor, only the residue R_S is affected by the fault on the sensor. In Figure 6.8, we can see that these residues corresponds to the fault alarm signature

$$\Sigma_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{6.71}$$

of the signature matrix **S** given in (6.68).

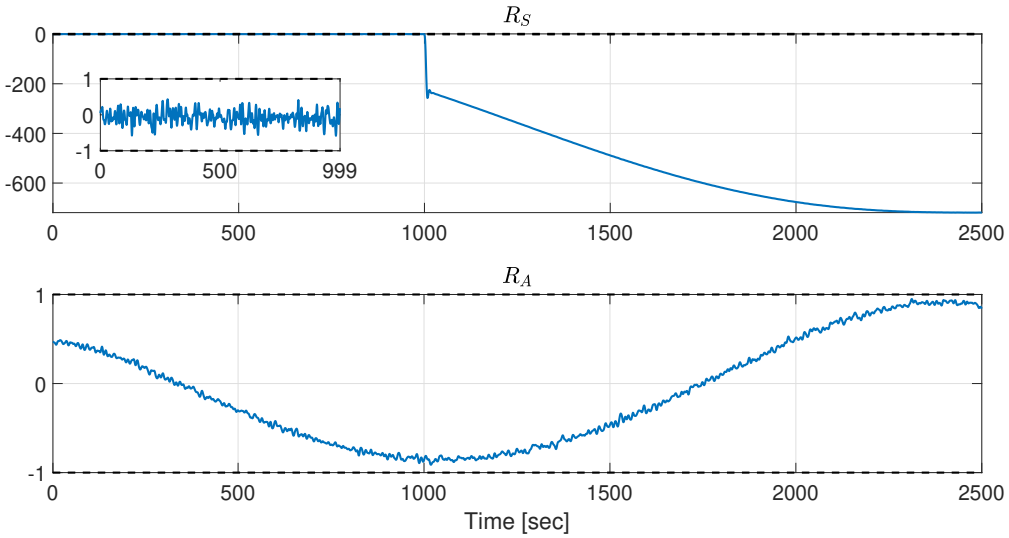

FIGURE 6.8: *Multiplicative fault on the sensor at time* $t = 1000s$

6.4.4.3 Multiplicative actuator fault

At time $t = 1000s$, we apply a multiplicative fault on the actuator, that is the actuator can provide only 80% of the reference trajectory of the heat flux $\varphi_{ref}(t)$. See Figure 6.9.

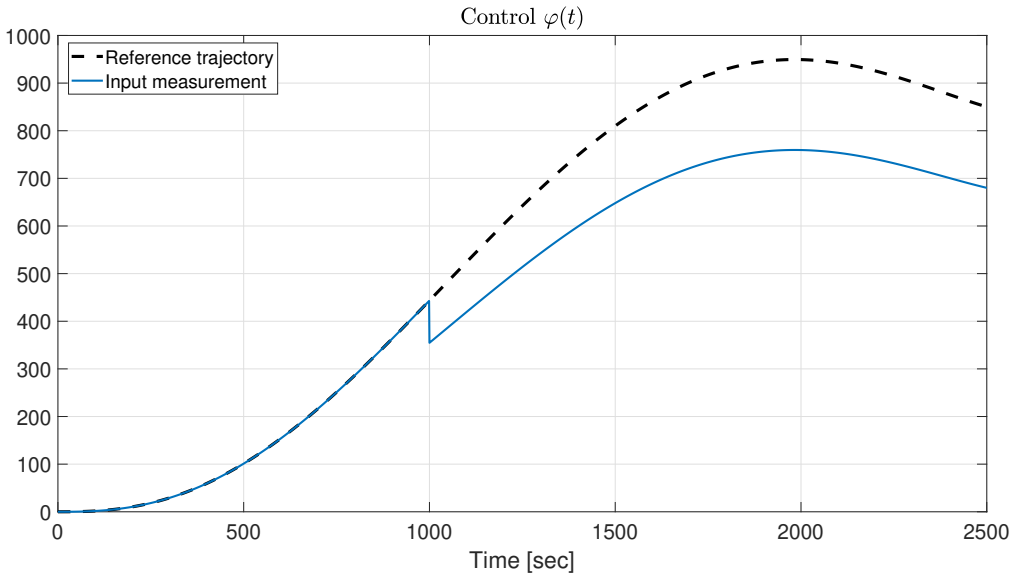

FIGURE 6.9: *Real heat flux vs. reference trajectory*

Moreover, the fault on the actuator affects the system states, including the flat output, and therefore affects the system output. See Figure 6.10.

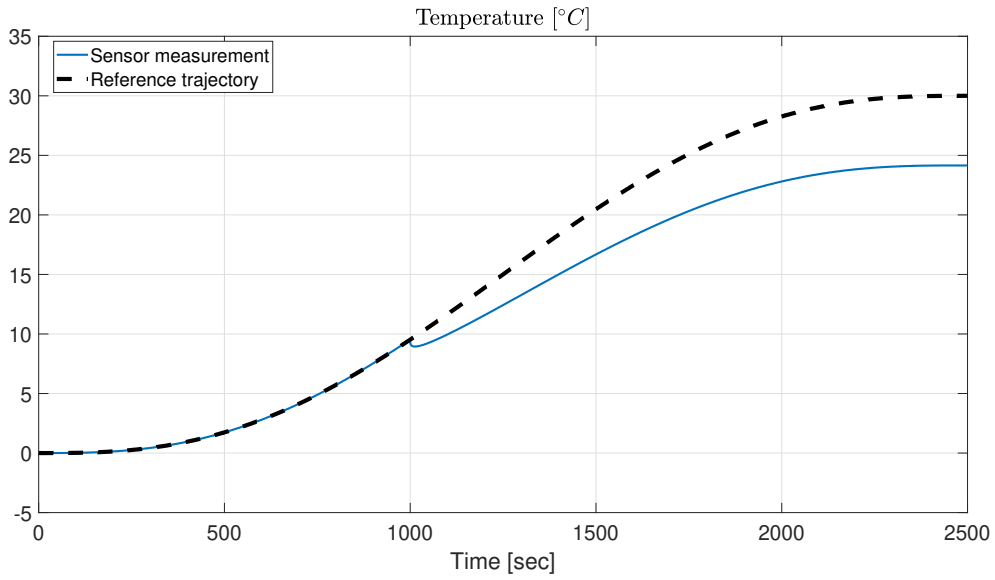

FIGURE 6.10: *Temperature measurement vs. reference trajectory after multiplicative fault on the actuator*

In order to construct the residue vector, we first compute the value of the flat output $Z^s(t)$ using relation (6.66). Then, we compute the redundant input $\varphi^z(t)$ and the redundant output $T^z(x_0, y_0, t)$ using (6.4) and (6.5).

Finally, the vector of residues can be computed, see Figure 6.11. We can see that a fault that affects the actuator, affects only the residue R_A, which corresponds to the fault alarm signature

$$\Sigma_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{6.72}$$

of the signature matrix **S** given in (6.68). Then, all multiplicative faults on the thermal bi-dimensional system can be detected and isolated using the fractional flatness-based FDI method presented in Section 6.3.

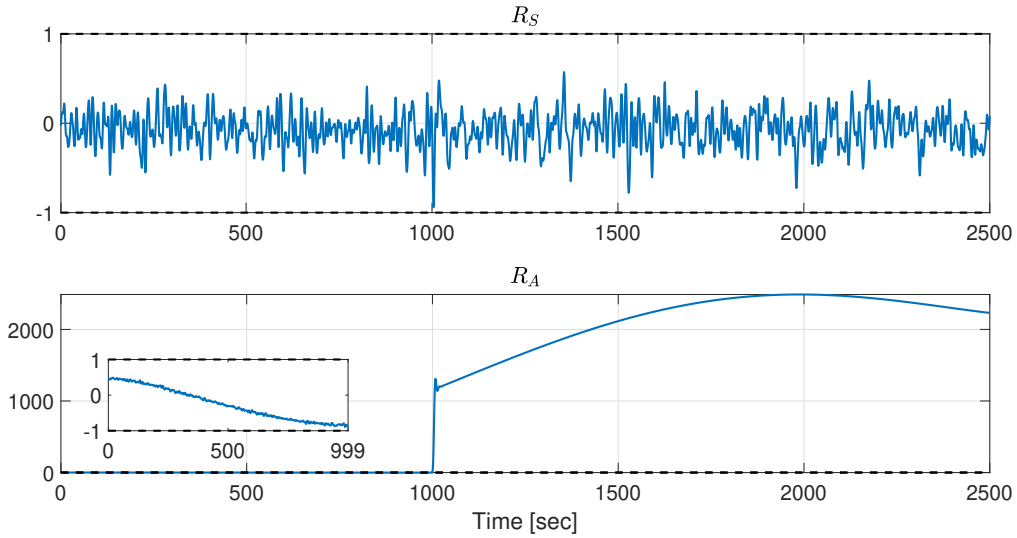

FIGURE 6.11: *Multiplicative fault on the actuator at time* $t = 1000s$

6.4.4.4 Additive faults

Additive faults on the sensor and the actuator are represented by biases. For the sensor fault, a $+2°C$ is added to the temperature at time $t = 1000s$, and for the actuator fault an extra heat flux of $10^2\ W/m^2$ is added to the heat flux. The residue values are illustrated in Figures 6.12 and 6.13.

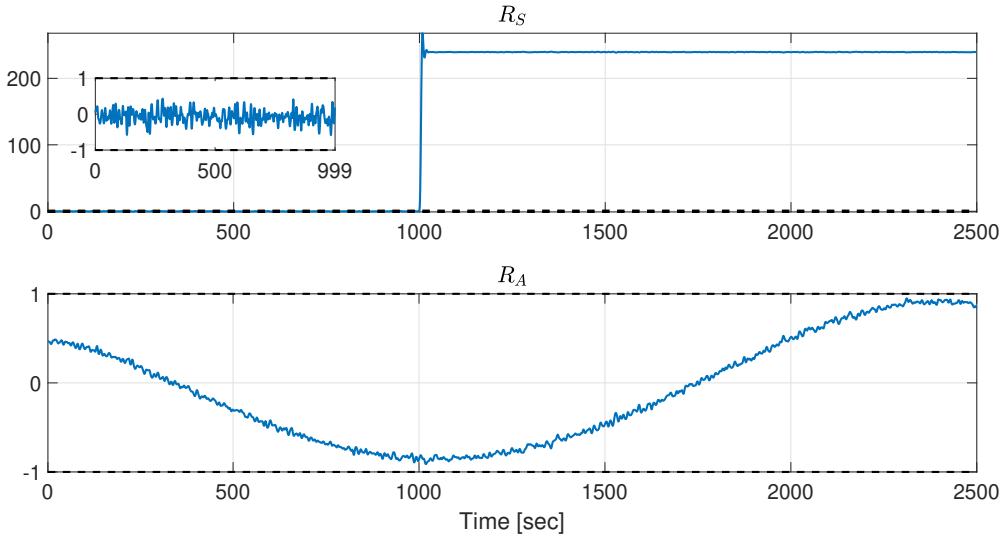

FIGURE 6.12: *Additive fault on the sensor at time* $t = 1000s$

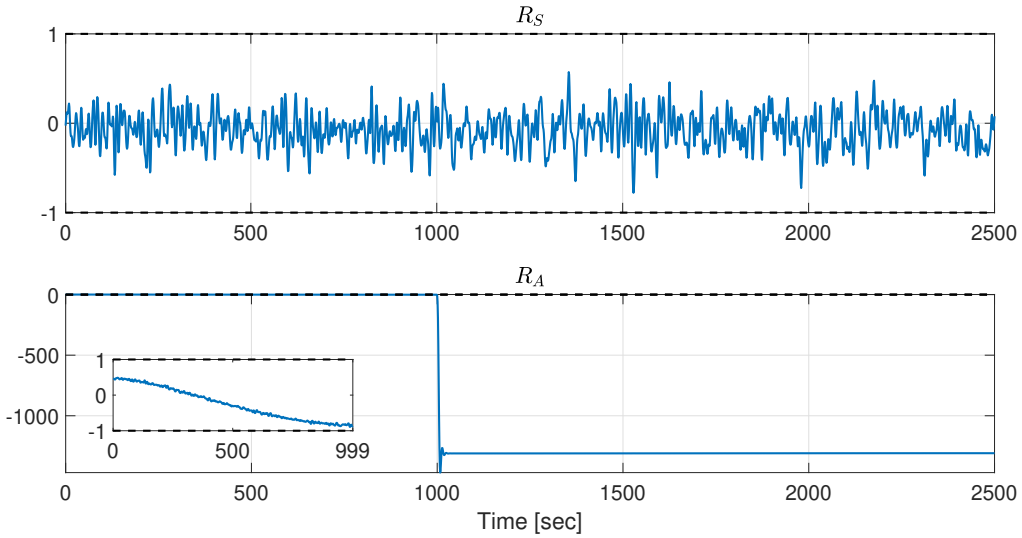

FIGURE 6.13: *Additive fault on the actuator at time* $t = 1000s$

Clearly, we have the same signature matrix obtained theoretically:

$$\mathbf{S} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{6.73}$$

Then, additive faults are also detectable and isolable using the fractional flatness-based FDI method.

6.5 Conclusion

In this chapter, the flatness-based FDI method, introduced in Martínez-Torres et al. (2014), and generalized in Chapter 3, has been extended to the class of fractional linear flat systems, in order to detect and isolate faults on sensors and actuators. In particular, definitions of residual generation, signature matrix, detectability and isolability were extended.

This new method of FDI has been applied on the thermal bi-dimensional system and its efficiency has been proved by simulations on an open-loop system.

This chapter presented a first work for the generalization of the flatness-based FDI method to the class of fractional linear systems. Future work must take into account the impact of a controller on the system, and apply it to a real thermal

system such as the thermal bar. Moreover, since the FDI process is a stage of the FTC process, a study can be made to show the effectiveness of the fractional flatness-based method on the control reconfiguration and fault tolerant control.

General Conclusions and Future Perspectives

Since the 1990s, the flatness property of dynamical systems has become a central topic in the field of automatic control theory (Levine, 2009; Sira-Ramirez and Agrawal, 2018). Numerous successful applications have been created, reflecting the importance of the differential flatness property in many areas such as robotics (Kiss, Levine, and Lantos, 1999), non-holonomic vehicles (Fliess et al., 1995a), aeronautic (Martin, Devasia, and Paden, 1996), hydraulic (Bindel et al., 2000) and automobile industry (Bitauld, Fliess, and Lévine, 1997; Lévine and Rémond, 2000). It has particularly demonstrated its efficiency for the generation and tracking of reference trajectories (Yang, Pan, and Wan, 2019; Antritter, Müller, and Deutscher, 2004), and designing robust controllers (Cazaurang, 1997; Lavigne, 2003).

Many methods of FDI have also been developed based on the flatness property. See Mai, Join, and Reger (2006), Nan et al. (2008), and Suryawan, De Doná, and Seron (2010). A novel flatness-based FDI method, introduced in Martínez-Torres et al. (2014), uses the flat outputs to construct the redundant variables in order to compare them to the real measurements of sensors and actuators. If at least one of the resulting residues exceeds its threshold then a fault is detected, otherwise no fault is detected. However, for the purpose of isolability, sometimes a single flat output is insufficient to isolate all faults, and a second or maybe multiple flat outputs are needed to improve the isolability. Importantly, despite the infinite number of flat outputs, the choice of flat outputs for fault isolation is not arbitrary, *i.e.* there are flat outputs that used together increase the isolability of faults and others that do not (Torres, 2014), hence the importance of the characterization of the flat outputs for the aim of FDI.

In Chapter 1, an overview on the FDI methods, that exist in the literature, is

provided with a particular attention to the methods based on the flatness property. Whereas, in Chapter 2, recalls on the flatness property of dynamic systems are presented with the application to the generation of reference trajectories.

In Chapter 3, the flatness-based FDI method, developed in Torres (2014), is presented with some generalizations. Particularly, we have introduced definitions for the detectability and isolability using the notions of signature matrix and fault alarm signature. Moreover, we proposed a characterization of the flat outputs for the aim of FDI that rigorously completes some heuristic results from Martínez-Torres et al. (2013b).

In Chapter 4, an application of the flat output characterization, proposed in Chapter 3, was performed on the three-tank system. In particular, we showed that the full isolability of faults is reachable by using two flat outputs that are independent. The effectiveness of this characterization has also been demonstrated by simulations on two cases: the open-loop system and the closed-loop system, in order to show how the addition of a controller can affect the generation of residual signals and then the isolability of faults. In addition to these simulations, experiments on the real three-tank system have been carried out.

These experiments have shown that this method can be sensitive to uncertainties and hence requires further development in the future. As far as we know, another limitation of this flatness-based FDI method is that there is no toolbox that is able to compute this infinity of flat outputs. The toolbox developed by Verhoeven (2016) was a big step, however, it still needs more developments and simplifications to be more user-friendly. In addition to that, for future works, the property of independence can be used to develop an algorithm able to directly compute the flat outputs that are useful for FDI and perhaps, can compute the number of the independent flat outputs that can be found for the considered system.

In the last decade, many studies have shown that there exist processes like thermal systems (Battaglia et al., 2000), viscoelastic systems (Moreau, Ramus-Serment, and Oustaloup, 2002) and nuclear magnetic resonance systems (Magin et al., 2008) that can be modeled by fractional differential equations. Therefore, fractional FDI methods needed to be developed (Aoun et al., 2011).

In Victor et al. (2015), the flatness property of integer-order dynamic systems has

been extended to include the class of fractional-order linear systems. It turns out that the fractional linear systems are flat if, and only if, they are controllable and the fractional flat output is the variable resulting from the Brunovský's canonical form (Victor, 2010). A method of computation of fractional flat outputs has been also developed based on the Smith decomposition of polynomial matrices with entries as polynomials having the fractional derivative operator $\mathbf{D}^{\gamma}$ as indeterminate.

In Chapter 5 of this thesis, another method of computation of fractional flat outputs, based on the unimodular completion algorithm of polynomial matrices, is developed. It is an extension of the unimodular completion algorithm developed in Fritzsche et al. (2016a) for the class of integer-order systems. In Chapter 6, the flatness-based FDI method, presented in Chapter 3, has been extended to the class of fractional linear flat systems. The redundant variables are computed using the measurements of the fractional flat outputs as long as theses measurements exist. This method was then applied to the thermal bi-dimensional system. In this system, using a single fractional flat output, faults on the sensor and the actuator were isolable. Moreover, we showed that this method is valid for both additive and multiplicative faults.

Fractional systems are generally characterized by a time delay. Then, in future works, an extension of the flatness property to the class of fractional time-delay linear systems can be developed, with an algorithm for computing the associated fractional flat outputs. In addition, the FDI method, based on fractional flatness, presented in Chapter 6, can also be extended to the class of fractional time-delay linear systems. As this method is based on the calculation of fractional flat outputs, a toolbox for the calculation of these can also be developed. Finally, the flatness property has been introduced originally to deal with nonlinear systems. As presented in Chapter 5, the extension of the flatness property to the class of fractional nonlinear systems is still not possible due to the absence of some necessary mathematical tools like differential calculus. Therefore, one of the main perspectives would be to extend differential calculus to fractional systems.

Bibliography

Antritter, Felix, Franck Cazaurang, Jean Lévine, and Johannes Middeke (2014). "On the computation of π-flat outputs for linear time-varying differential-delay systems". In: *Systems & Control Letters* 71, pp. 14–22.

Antritter, Felix and Johannes Middeke (2011). "A toolbox for the analysis of linear systems with delays". In: *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, pp. 1950–1955.

Antritter, Felix, Bernd Müller, and Joachim Deutscher (2004). "Tracking control for nonlinear flat systems by linear dynamic output feedback". In: *IFAC Proceedings Volumes* 37.13, pp. 141–146.

Aoun, M, A Aribi, S Najar, and MN Abdelkrim (2010). "On the fractional systems' fault detection". In: *4th IFAC symposium on fractional differentiation and its applications (FDA)*, pp. 1–6.

Aoun, Mohamed, Asma Aribi, Slaheddine Najar, and Mohamed Naceur Abdelkrim (2011). "On the fractional systems' fault detection: A comparison between fractional and rational residual sensitivity". In: *Eighth International Multi-Conference on Systems, Signals & Devices*. IEEE, pp. 1–6.

Aribi, Asma, Mohamed Aoun, Christophe Farges, Pierre Melchior, Slaheddine Najar, and Mohamed Naceur Abdelkrim (2012a). "Robust dynamic parity space method for fractional order systems fault detection". In:

Aribi, Asma, Mohamed Aoun, Christophe Farges, Slaheddine Najar, Pierre Melchior, and Mohamed Naceur Abdelkrim (2013). "Generalied fractional obsevers scheme to fault detection and isolation". In: *10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13)*. IEEE, pp. 1–7.

Aribi, Asma, Mohamed Aoun, Slaheddine Najar, and Mohamed Naceur Abdelkrim (2012b). "Evaluation of fractional residual". In: *International Multi-Conference on Systems, Signals & Devices*. IEEE, pp. 1–6.

Baker, George A, George A Baker Jr, George Baker, Peter Graves-Morris, and Susan S Baker (1996). *Pade Approximants: Encyclopedia of Mathematics and It's Applications, Vol. 59 George A. Baker, Jr., Peter Graves-Morris*. Vol. 59. Cambridge University Press.

Battaglia, Jean-Luc, Ludovic Le Lay, Jean-Christophe Batsale, Alain Oustaloup, and Olivier Cois (2000). "Utilisation de modèles d'identification non entiers pour la résolution de problèmes inverses en conduction". In: *International journal of thermal sciences* 39.3, pp. 374–389.

Belghith, Nader, Michel Fliess, François Ollivier, and Alexandre Sedoglavic (2003). "Platitude différentielle de l'équation de la chaleur bidimensionnelle". In: *Actes des journées doctorales d'automatique. Valenciennes, France. June*, pp. 25–27.

Bindel, R, R Nitsche, R Rothfuß, and Michael Zeitz (2000). "Flachheitsbasierte Regelung eines hydraulischen Antriebs mit zwei Ventilen für einen Großmanipulator (Flatness Based Control of a Two Valve Hydraulic Joint of a Large Manipulator)". In: *at-Automatisierungstechnik* 48.3, p. 124.

Bitauld, Lionel, Michel Fliess, and Jean Lévine (1997). "A flatness based control synthesis of linear systems and application to windshield wipers". In: *1997 European Control Conference (ECC)*. IEEE, pp. 2460–2465.

Blanke, Mogens, Michel Kinnaert, Jan Lunze, Marcel Staroswiecki, and Jochen Schröder (2006). *Diagnosis and fault-tolerant control*. Vol. 2. Springer.

Bocharov, AV, IS Krasil Shchik, and Aleksandr Mikhaĭlovich Vinogradov (1999). *Symmetries and conservation laws for differential equations of mathematical physics*. Vol. 182. American Mathematical Society Providence, RI.

Brunovskỳ, Pavol (1970). "A classification of linear controllable systems". In: *Kybernetika* 6.3, pp. 173–188.

Cazaurang, Franck (1997). "Commande robuste des systèmes plats: application à la commande d'une machine synchrone." PhD thesis.

Chen, Jianhong, Hongkun Li, Deren Sheng, and Wei Li (2015). "A hybrid data-driven modeling method on sensor condition monitoring and fault diagnosis for power plants". In: *International Journal of Electrical Power & Energy Systems* 71, pp. 274–284.

Chen, Jie and Ron J Patton (2012). *Robust model-based fault diagnosis for dynamic systems*. Vol. 3. Springer Science & Business Media.

Chow, EYEY and Alan Willsky (1984). "Analytical redundancy and the design of robust failure detection systems". In: *IEEE Transactions on automatic control* 29.7, pp. 603–614.

Cohn, Paul Moritz (1971). *Free rings and their relations*.

Cottrill-Shepherd, Kathleen and Mark Naber (2001). "Fractional differential forms". In: *Journal of Mathematical Physics* 42.5, pp. 2203–2212.

Crassidis, John L and John L Junkins (2011). *Optimal estimation of dynamic systems*. CRC press.

Dexter, Arthur L and D Ngo (2001). "Fault diagnosis in air-conditioning systems: a multi-step fuzzy model-based approach". In: *HVAC&R Research* 7.1, pp. 83–102.

Ding, Steven X (2008). *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Science & Business Media.

— (2012). "A survey of fault-tolerant networked control system design". In: *IFAC Proceedings Volumes* 45.20, pp. 874–885.

Dugowson, Stéphane (1994). "Les différentielles métaphysiques: histoire et philosophie de la généralisation de l'ordre de la dérivation". PhD thesis. Paris 13.

Euler, Leonhard (1738). "De progressionibus transcendentibus seu quarum termini generales algebraice dari nequeunt". In: *Commentarii academiae scientiarum Petropolitanae*, pp. 36–57.

Fliess, M, J Lévine, P Martin, and P Rouchon (1992). "On Differentially Flat Nonlinear Systems". In: *IFAC Proceedings Volumes* 25.13, pp. 159–163.

— (1995a). "Design of trajectory stabilizing feedback for driftless at systems". In: *Proceedings of the Third ECC, Rome*, pp. 1882–1887.

Fliess, Michel and Richard Hotzel (1997). "On linear systems with a derivation of non-integer order". In: *Comptes Rendus de l'Academie des Sciences Series IIB Mechanics Physics Chemistry Astronomy* 2.324, pp. 99–105.

Fliess, Michel, Jean Lévine, Philippe Martin, and Pierre Rouchon (1993). "On differentially flat nonlinear systems". In: *Nonlinear Control Systems Design 1992*. Elsevier, pp. 159–163.

Fliess, Michel, Jean Lévine, Philippe Martin, and Pierre Rouchon (1995b). "Flatness and defect of non-linear systems: introductory theory and examples". In: *International journal of control* 61.6, pp. 1327–1361.

— (1999). "A lie-backlund approach to equivalence and flatness of nonlinear systems". In: *IEEE Transactions on automatic control* 44.5, pp. 922–937.

Fourier, Jean Baptiste Joseph baron (1822). *Théorie analytique de la chaleur*. F. Didot.

Frank, Paul M (1990). "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results". In: *automatica* 26.3, pp. 459–474.

Frank, Paul M and Xianchun Ding (1997). "Survey of robust residual generation and evaluation methods in observer-based fault detection systems". In: *Journal of process control* 7.6, pp. 403–424.

Frank, PM, G Schrier, and E Alcorta Garcia (1999). "Nonlinear observers for fault detection and isolation". In: *New Directions in nonlinear observer design*. Springer, pp. 399–422.

Franke, Matthias and Klaus Röbenack (2013). "On the computation of flat outputs for nonlinear control systems". In: *2013 European Control Conference (ECC)*. IEEE, pp. 167–172.

Fritzsche, Klemens (2016). *Implementation of an algorithm for computing unimodular completions of hyperregularmatrices*. https://github.com/klim-/uc_algorithm.

Fritzsche, Klemens, Matthias Franke, Carsten Knoll, and Klaus Röbenack (2016a). "Zur systematischen Bestimmung flacher Ausgänge nichtlinearer Mehrgrößensysteme". In: *at-Automatisierungstechnik* 64.12, pp. 948–960.

Fritzsche, Klemens, Carsten Knoll, Matthias Franke, and Klaus Röbenack (2016b). "Unimodular completion and direct flat representation in the context of differential flatness". In: *PAMM* 16.1, pp. 807–808.

Gao, Zhiwei, Carlo Cecati, and Steven X Ding (2015). "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches". In: *IEEE Transactions on Industrial Electronics* 62.6, pp. 3757–3767.

Gertler, Janos and David Singer (1990). "A new structural framework for parity equation-based failure detection and isolation". In: *Automatica* 26.2, pp. 381–388.

Gertler, Janos J (1988). "Survey of model-based failure detection and isolation in complex plants". In: *IEEE Control systems magazine* 8.6, pp. 3–11.

Group, IEEE Working et al. (1992). "Recommended practice for excitation system models for power system stability studies". In: *IEEE Std 421.5-1992*.

Kailath, Thomas (1980). *Linear systems*. Vol. 156. Prentice-Hall Englewood Cliffs, NJ.

Kalman, Rudolf Emil (1963). "Mathematical description of linear dynamical systems". In: *Journal of the Society for Industrial and Applied Mathematics, Series A: Control* 1.2, pp. 152–192.

Kalman, Rudolf Emil et al. (1960). "Contributions to the theory of optimal control". In: *Bol. soc. mat. mexicana* 5.2, pp. 102–119.

Kaminski, Yirmeyahu J, Jean Lévine, and François Ollivier (2018). "Intrinsic and apparent singularities in differentially flat systems, and application to global motion planning". In: *Systems & Control Letters* 113, pp. 117–124.

Khan, Abdul Qayyum and Steven X Ding (2009). "Threshold computation for robust fault detection in a class of continuous-time nonlinear systems". In: *2009 European Control Conference (ECC)*. IEEE, pp. 3088–3093.

— (2011). "Threshold computation for fault detection in a class of discrete-time nonlinear systems". In: *International journal of adaptive control and signal processing* 25.5, pp. 407–429.

Kiss, Balint, Jean Levine, and Bela Lantos (1999). "Trajectory planning for dextrous manipulation with rolling contacts". In: *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304)*. Vol. 3. IEEE, pp. 2118–2119.

Kolchin, Ellis Robert (1973). *Differential algebra & algebraic groups*. Academic press.

Kóscielny, Jan Maciej, Michał Syfert, Kornel Rostek, and Anna Sztyber (2016). "Fault isolability with different forms of the faults–symptoms relation". In: *International Journal of Applied Mathematics and Computer Science* 26.4, pp. 815–826.

Krasilchchik, I, Aleksandr Michajlovič Vinogradov, and VV Lychagin (1996). *Geometry of jet spaces and nonlinear partial differential equations*. Gordon and Breach.

Latombe, Jean-Claude (1991). "Robot Motion Planning". In:

LaValle, Steven M (2006). *Planning algorithms*. Cambridge university press.

Lavigne, Loïc (2003). "Outils d'analyse et de synthèse des lois de commande robuste des systèmes dynamiques plats." PhD thesis.

Lévine, J (1999). "Are there new industrial perspectives in the control of mechanical systems?" In: *Advances in Control*. Springer, pp. 197–226.

Lévine, Jean (2006). "On Necessary and Sufficient Conditions for Differential Flatness". In: *arXiv preprint math/0605405*.

Levine, Jean (2009). *Analysis and control of nonlinear systems: A flatness-based approach*. Springer Science & Business Media.

Lévine, Jean (2011). "On necessary and sufficient conditions for differential flatness". In: *Applicable Algebra in Engineering, Communication and Computing* 22.1, pp. 47–90.

Levine, Jean, Jacques Lottin, and J-C Ponsart (1996). "A nonlinear approach to the control of magnetic bearings". In: *IEEE transactions on control systems technology* 4.5, pp. 524–544.

Lévine, Jean and DV Nguyen (2003). "Flat output characterization for linear systems using polynomial matrices". In: *Systems & control letters* 48.1, pp. 69–75.

Lévine, Jean and Bernard Rémond (2000). "Flatness based control of an automatic clutch". In: *Proc. MTNS-2000, Perpignan*. Vol. 70, pp. 1995–2000.

Liouville, Joseph (1832). *Mémoire sur quelques questions de géométrie et de mécanique, et sur un nouveau genre de calcul pour résoudre ces questions*.

López-Fernandez, JL and C Raventós Olivella (2009). "Model based fault detection using differential flatness". In: *5th AIAA-PEGASUS Student conference, Toulouse, France*.

Magin, Richard L, Osama Abdullah, Dumitru Baleanu, and Xiaohong Joe Zhou (2008). "Anomalous diffusion expressed through fractional order differential operators in the Bloch–Torrey equation". In: *Journal of Magnetic Resonance* 190.2, pp. 255–270.

Mai, Philipp, Cédric Join, and Johan Reger (2006). "An example of flatness based fault tolerant control using algebraic derivative estimation". In: *IAR Workshop, Nancy*.

— (2007). "Flatness-based fault tolerant control of a nonlinear MIMO system using algebraic derivative estimation". In: *IFAC Proceedings Volumes* 40.20, pp. 350–355.

Marquez, R and E Delaleau (1999). "Une application de la commande prédictive linéaire basée sur la platitude". In: *Actes Journées Doctorales d'Automatique*, pp. 148–152.

Martin, Philippe (1992). "Contribution à l'étude des systèmes différentiellement plats". PhD thesis.

Martin, Philippe, Santosh Devasia, and Brad Paden (1996). "A different look at output tracking: control of a VTOL aircraft". In: *Automatica* 32.1, pp. 101–107.

Martínez-Torres, César, Loïc Lavigne, Franck Cazaurang, Efraín Alcorta-García, and David A Díaz-Romero (2013a). "Fault detection and isolation on a three tank system using differential flatness". In: *2013 European Control Conference (ECC)*. IEEE, pp. 2433–2438.

— (2013b). "Fault tolerant control of a three tank system: A flatness based approach". In: *2013 Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, pp. 529–534.

— (2014). "Flatness-based fault tolerant control". In: *Dyna* 81.188, pp. 131–138.

Matignon, Denis and Brigitte d'Andréa Novel (1996). "Some results on controllability and observability of finite-dimensional fractional differential systems". In: *Computational engineering in systems applications*. Vol. 2. Citeseer, pp. 952–956.

Mboup, Mamadou, Cédric Join, and Michel Fliess (2007). "A revised look at numerical differentiation with an application to nonlinear feedback control". In: *2007 Mediterranean Conference on Control & Automation*. IEEE, pp. 1–6.

Miller, Kenneth S and Bertram Ross (1993). *An introduction to the fractional calculus and fractional differential equations*. Wiley.

Moreau, Xavier, Caroline Ramus-Serment, and Alain Oustaloup (2002). "Fractional differentiation in passive vibration control". In: *Nonlinear Dynamics* 29.1-4, pp. 343–362.

Nan, Zhang, Drouin Antoine, Doncescu Andrei, and Mora-Camin Felixo (2008). "A differential flatness approach for rotorcraft fault detection". In: *2008 27th Chinese Control Conference*. IEEE, pp. 237–241.

NøRgaard, Magnus, Niels K Poulsen, and Ole Ravn (2000). "New developments in state estimation for nonlinear systems". In: *Automatica* 36.11, pp. 1627–1638.

Noura, Hassan, Didier Theilliol, Jean-Christophe Ponsart, and Abbas Chamseddine (2009). *Fault-tolerant control systems: Design and practical applications*. Springer Science & Business Media.

Oustaloup, A, P Melchior, P Lanusse, O Cois, and F Dancla (2000). "The CRONE toolbox for Matlab". In: *CACSD. Conference Proceedings. IEEE International Symposium on Computer-Aided Control System Design (Cat. No. 00TH8537)*. IEEE, pp. 190–195.

Oustaloup, Alain (1995). *La dérivation non entière*. BOOK. Hermes.

Patan, Krzysztof and Thomas Parisini (2005). "Identification of neural dynamic models for fault detection and isolation: the case of a real sugar evaporation process". In: *Journal of Process Control* 15.1, pp. 67–79.

Patton, Ron (1993). "Robustness issues in fault-tolerant control". In: *IEE Colloquium on Fault Diagnosis and Control System Reconfiguration*. IET, pp. 1–1.

Patton, Ron J (1997). "Fault-tolerant control: the 1997 situation". In: *IFAC Proceedings Volumes* 30.18, pp. 1029–1051.

Petit, N (2000). "Platitude et planification de trajectoires pour certains systèmes à retards et EDP: applications en génie chimique". In: *These de doctorat, Ecole des Mines de Paris, Paris*.

Podlubny, Igor (1999). *Fractional differential equations, vol. 198 of Mathematics in Science and Engineering*.

Pomet, J-B (1997). "On dynamic feedback linearization of four-dimensional affine control systems with two inputs". In: *ESAIM: Control, optimisation and calculus of variations* 2, pp. 151–230.

Raie, A and V Rashtchi (2002). "Using a genetic algorithm for detection and magnitude determination of turn faults in an induction motor". In: *Electrical Engineering* 84.5, pp. 275–279.

Rao, Ming, Qijun Xia, and Yiqun Ying (2013). *Modeling and advanced control for process industries: applications to paper making processes*. Springer Science & Business Media.

Rato, L and JM Lemos (1999). "Multimodel based fault tolerant control of the 3-tank system". In: *1999 European Control Conference (ECC)*. IEEE, pp. 4047–4052.

Reif, John H (1979). "Complexity of the mover's problem and generalizations". In: *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*. IEEE, pp. 421–427.

Riemann, Bernhard, Richard Dedekind, and Heinrich Weber (1892). *Gesammelte mathematische Werke und wissenschaftlicher Nachlass*.

Rincon-Pasaye, J Juan, Rafael Martinez-Guerra, and Alberto Soria-Lopez (2008). "Fault diagnosis in nonlinear systems: An application to a three-tank system". In: *2008 American Control Conference*. IEEE, pp. 2136–2141.

Ritt, Joseph Fels (1950). *Differential algebra*. Vol. 33. American Mathematical Soc.

Rothfuss, Ralf, J Rudolph, and Michael Zeitz (1996). "Flatness based control of a nonlinear chemical reactor model". In: *Automatica* 32.10, pp. 1433–1439.

Rudolph, Joachim (2003). *Flatness based control of distributed parameter systems:[notes for a course at the Max Planck Institute for dynamics of complex technical systems at Magdeburg, Germany, on February 24-28, 2003]*. Shaker.

Samko, Stefan G, Anatoly A Kilbas, Oleg I Marichev, et al. (1993). *Fractional integrals and derivatives*. Vol. 1. Gordon and Breach Science Publishers, Yverdon Yverdon-les-Bains, Switzerland.

Sira-Ramirez, Hebertt and Sunil K Agrawal (2018). *Differentially flat systems*. Crc Press.

Stigler, Stephen M (2005). "PS Laplace, Théorie analytique des probabilités, (1812); Essai philosophique sur les probabilités, (1814)". In: *Landmark Writings in Western Mathematics 1640-1940*. Elsevier, pp. 329–340.

Stumper, Jean-François, Ferdinand Svaricek, and Ralph Kennel (2009). "Trajectory tracking control with flat inputs and a dynamic compensator". In: *2009 European Control Conference (ECC)*. IEEE, pp. 248–253.

Suryawan, Fajar, José De Doná, and María Seron (2010). "Fault detection, isolation, and recovery using spline tools and differential flatness with application to a magnetic levitation system". In: *2010 Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, pp. 293–298.

Theilliol, Didier, Hassan Noura, and Jean-Christophe Ponsart (2002). "Fault diagnosis and accommodation of a three-tank system based on analytical redundancy". In: *ISA transactions* 41.3, pp. 365–382.

Thirumarimurugan, M, N Bagyalakshmi, and P Paarkavi (2016). "Comparison of fault detection and isolation methods: A review". In: *2016 10th International Conference on Intelligent Systems and Control (ISCO)*. IEEE, pp. 1–6.

Torres, César Martínez (2014). "Fault Tolerant Control By Flatness Approach". PhD thesis. Université de Bordeaux.

Trigeassou, Jean-Claude and Nezha Maamri (2019). *Analysis, Modeling and Stability of Fractional Order Differential Systems 2: The Infinite State Approach*. John Wiley & Sons.

Vasiljevic, Luma K and Hassan K Khalil (2008). "Error bounds in differentiation of noisy signals by high-gain observers". In: *Systems & Control Letters* 57.10, pp. 856–862.

Venkatasubramanian, Venkat, Raghunathan Rengaswamy, Surya N Kavuri, and Kewen Yin (2003). "A review of process fault detection and diagnosis: Part III: Process history based methods". In: *Computers & chemical engineering* 27.3, pp. 327–346.

Verhoeven, Gregor Günther (2016). "Symbolic Software Tools for Flatness of Linear Systems with Delays and Nonlinear Systems". PhD thesis. Universitätsbibliothek der Universität der Bundeswehr München.

Victor, Stéphane (2010). "Identification par modèle non entier pour la poursuite robuste de trajectoire par platitude". PhD thesis.

Victor, Stéphane, Pierre Melchior, Jean Lévine, and Alain Oustaloup (2015). "Flatness for linear fractional systems with application to a thermal system". In: *Automatica* 57, pp. 213–221.

Wu, Lifeng, Beibei Yao, Zhen Peng, and Yong Guan (2017). "An adaptive threshold algorithm for sensor fault based on the grey theory". In: *Advances in Mechanical Engineering* 9.2, p. 1687814017693193.

Yang, Yajue, Jia Pan, and Weiwei Wan (2019). "Survey of optimal motion planning". In: *IET Cyber-systems and Robotics* 1.1, pp. 13–19.

Zehetner, Josef, Johann Reger, and Martin Horn (2007). "A derivative estimation toolbox based on algebraic methods-theory and practice". In: *2007 IEEE International Conference on Control Applications*. IEEE, pp. 331–336.

Zhang, Nan, Andrei Doncescu, Alexandre Carlos Brandao-Ramos, and Felix Mora-Camino (2012). "Fault detection for difference flat systems". In:

Zhang, Youmin and Jin Jiang (2008). "Bibliographical review on reconfigurable fault-tolerant control systems". In: *Annual reviews in control* 32.2, pp. 229–252.

Zhou, Kemin, John Comstock Doyle, Keith Glover, et al. (1996). *Robust and optimal control*. Vol. 40. Prentice hall New Jersey.

Zhou, Yimin, Jingjing Liu, and Arthur L Dexter (2014). "Estimation of an incipient fault using an adaptive neurofuzzy sliding-mode observer". In: *Energy and buildings* 77, pp. 256–269.

Zhou, Yimin, Guoqing Xu, and Qi Zhang (2014). "Overview of fault detection and identification for non-linear dynamic systems". In: *2014 IEEE International Conference on Information and Automation (ICIA)*. IEEE, pp. 1040–1045.

Author Bibliography

Communications in international congresses

– Rammal, Rim and Airimitoaie, Tudor-Bogdan and Cazaurang, Franck and Lévine, Jean and Melchior, Pierre (2020). "On the Choice of Multiple Flat Outputs for Fault Detection and Isolation of a Flat System". In: *21th World Congress of the International Federation of Automatic Control*. preprint pdf.

– Rammal, Rim and Airimitoaie, Tudor-Bogdan and Melchior, Pierre and Cazaurang, Franck (2020). "Unimodular Completion for Computation of Fractionally Flat Outputs for Linear Fractionally Flat Systems". In: *21th World Congress of the International Federation of Automatic Control*. preprint pdf.

Communications in national congresses

– Rammal, Rim and Airimitoaie, Tudor-Bogdan and Cazaurang, Franck and Melchior, Pierre (2019). "Characterization of flat output sets using a symbolic calculation tool". In: *Journées Doctorales GDR MACS – 5 / 7 juin 2019*.